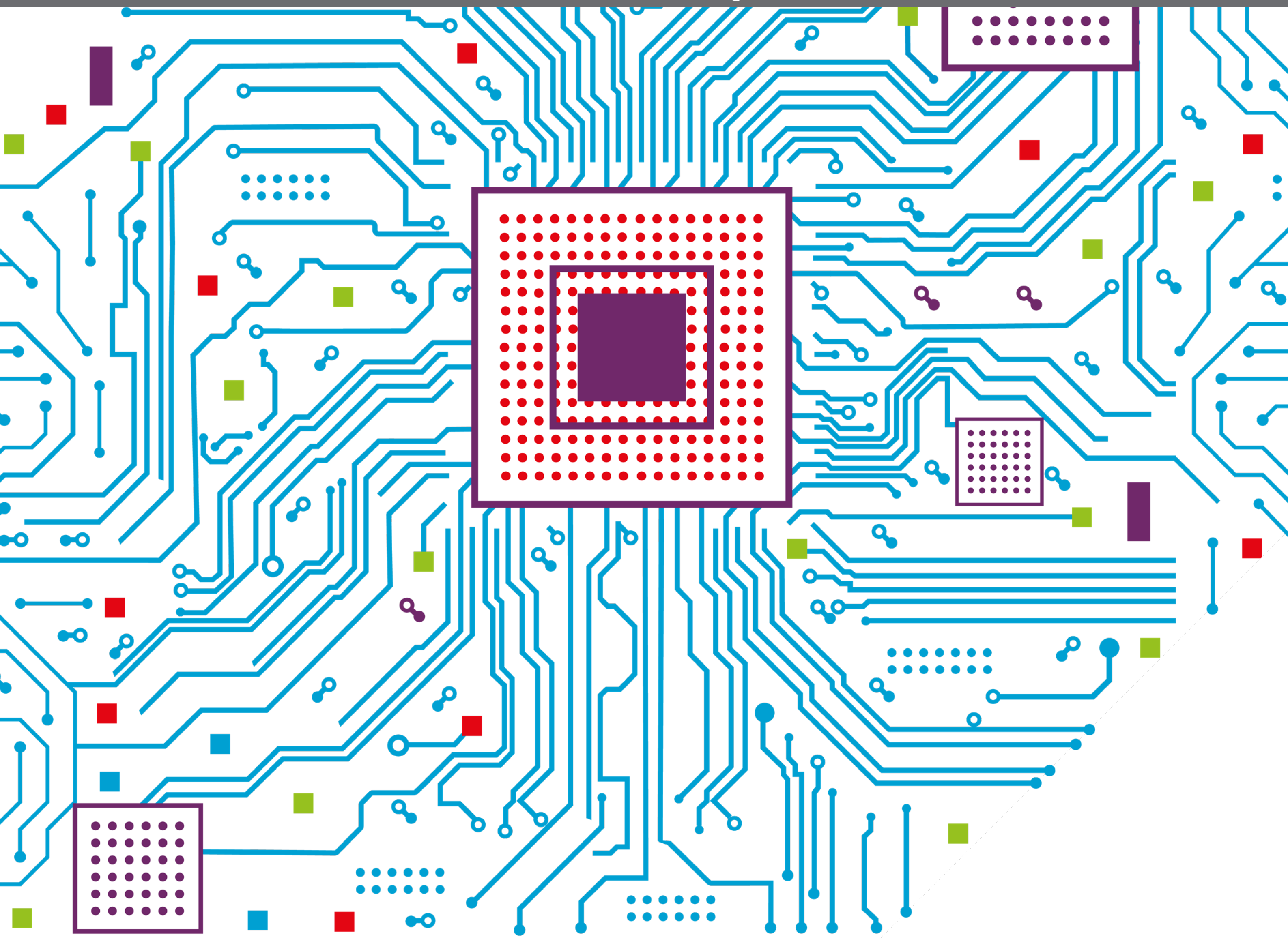
DATA-DRIVEN COGNITIVE MANUFACTURING - APPLICATIONS IN PREDICTIVE MAINTENANCE AND ZERO DEFECT MANUFACTURING

EDITED BY: Dimitris Kiritsis, Melinda Hodkiewicz, Oscar Lazaro, Jay Lee and Jun Ni

PUBLISHED IN: Frontiers in Computer Science, Frontiers in Mechanical Engineering and Frontiers in Artificial Intelligence



frontiers Research Topics

ISSN 1664-8714
ISBN 978-2-88966-583-9
DOI 10.3389/978-2-88966-583-9

About Frontiers

Frontiers is more than just an open-access publisher of scholarly articles: it is a pioneering approach to the world of academia, radically improving the way scholarly research is managed. The grand vision of Frontiers is a world where all people have an equal opportunity to seek, share and generate knowledge. Frontiers provides immediate and permanent online open access to all its publications, but this alone is not enough to realize our grand goals.

Frontiers Journal Series

The Frontiers Journal Series is a multi-tier and interdisciplinary set of open-access, online journals, promising a paradigm shift from the current review, selection and dissemination processes in academic publishing. All Frontiers journals are driven by researchers for researchers; therefore, they constitute a service to the scholarly community. At the same time, the Frontiers Journal Series operates on a revolutionary invention, the tiered publishing system, initially addressing specific communities of scholars, and gradually climbing up to broader public understanding, thus serving the interests of the lay society, too.

Dedication to Quality

Each Frontiers article is a landmark of the highest quality, thanks to genuinely collaborative interactions between authors and review editors, who include some of the world's best academicians. Research must be certified by peers before entering a stream of knowledge that may eventually reach the public - and shape society; therefore, Frontiers only applies the most rigorous and unbiased reviews.
Frontiers revolutionizes research publishing by freely delivering the most outstanding research, evaluated with no bias from both the academic and social point of view. By applying the most advanced information technologies, Frontiers is catapulting scholarly publishing into a new generation.

What are Frontiers Research Topics?

Frontiers Research Topics are very popular trademarks of the Frontiers Journals Series: they are collections of at least ten articles, all centered on a particular subject. With their unique mix of varied contributions from Original Research to Review Articles, Frontiers Research Topics unify the most influential researchers, the latest key findings and historical advances in a hot research area! Find out more on how to host your own Frontiers Research Topic or contribute to one as an author by contacting the Frontiers Editorial Office: frontiersin.org/about/contact

DATA-DRIVEN COGNITIVE MANUFACTURING - APPLICATIONS IN PREDICTIVE MAINTENANCE AND ZERO DEFECT MANUFACTURING

Topic Editors:
Dimitris Kiritsis, École Polytechnique Fédérale de Lausanne, Switzerland
Melinda Hodkiewicz, University of Western Australia, Australia
Oscar Lazaro, Innovalia Association, Spain
Jay Lee, University of Cincinnati, United States
Jun Ni, University of Michigan, United States

Citation: Kiritsis, D., Hodkiewicz, M., Lazaro, O., Lee, J., Ni, J., eds. (2021). Data-Driven Cognitive Manufacturing - Applications in Predictive Maintenance and Zero Defect Manufacturing. Lausanne: Frontiers Media SA. doi: 10.3389/978-2-88966-583-9

Table of Contents

04 ***Editorial: Data-Driven Cognitive Manufacturing—Applications in Predictive Maintenance and Zero Defect Manufacturing***
Dimitris Kiritsis, Oscar Lazaro, Melinda Hodkiewicz, Jay Lee and Jun Ni

06 ***Product Quality Improvement Policies in Industry 4.0: Characteristics, Enabling Factors, Barriers, and Evolution Toward Zero Defect Manufacturing***
Foivos Psarommatis, Sylvain Prouvost, Gökan May and Dimitris Kiritsis

21 ***Physically Inspired Data Compression and Management for Industrial Data Analytics***
Ramin Sabbagh, Zicheng Cai, Alec Stothert and Dragan Djurdjanovic

34 ***Ontology-Based Context Modeling in Physical Asset Integrity Management***
Ali Al-Shdifat, Christos Emmanouilidis, Muhammad Khan and Andrew G. Starr

50 ***Prognostics and Health Management of Industrial Assets: Current Progress and Road Ahead***
Luca Biggio and Iason Kastanis

74 ***Predictive Maintenance for Injection Molding Machines Enabled by Cognitive Analytics for Industry 4.0***
Vaia Rousopoulou, Alexandros Nizamis, Thanasis Vafeiadis, Dimosthenis Ioannidis and Dimitrios Tzovaras

86 ***Implementation and Transfer of Predictive Analytics for Smart Maintenance: A Case Study***
Sebastian von Enzberg, Athanasios Naskos, Ifigeneia Metaxa, Daniel Köchling and Arno Kühn

99 ***Intelligent Predictive Maintenance and Remote Monitoring Framework for Industrial Equipment Based on Mixed Reality***
Dimitris Mourtzis, John Angelopoulos and Nikos Panopoulos

111 ***RECLAIM: Toward a New Era of Refurbishment and Remanufacturing of Industrial Equipment***
Angeliki Zacharaki, Thanasis Vafeiadis, Nikolaos Kolokas, Aikaterini Vaxevani, Yuchun Xu, Michael Peschl, Dimosthenis Ioannidis and Dimitrios Tzovaras

frontiers in Computer Science

EDITORIAL
published: 08 February 2021
doi: 10.3389/fcomp.2020.633850

Editorial: Data-Driven Cognitive Manufacturing—Applications in Predictive Maintenance and Zero Defect Manufacturing

Dimitris Kiritsis[1], Oscar Lazaro[2], Melinda Hodkiewicz[3], Jay Lee[4] and Jun Ni[5]*

[1]École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, [2]Innovalia Association, Bizkaia, Spain, [3]University of Western Australia, Perth, WA, Australia, [4]University of Cincinnati, Cincinnati, OH, United States, [5]University of Michigan, Ann Arbor, MI, United States

Keywords: big data for factories, closed loop lifecycle management, zero defect manufacturing, predictive maintenance, semantic technologies, industrial ontologies, industrial AI, artificial intelligence

Editorial on the Research Topic

Data-Driven Cognitive Manufacturing - Applications in Predictive Maintenance and Zero Defect Manufacturing

Closed-Loop Lifecycle Management (CL2M) is an integral part of the circular economy. Managing the CL2M enables manufacturers and associated digital factories to connect in-service issues back to process conditions and product information at manufacturing and other stages of the life cycle with the aim of having Zero Defect Manufacturing (ZDM).

ZDM can be implemented through two approaches: product-oriented and process-oriented ZDM. Product-oriented ZDM studies defects in the actual parts., Process-oriented ZDM studies defects in the manufacturing equipment that have led, or might lead to product defects this is implemented through Predictive Maintenance.

The Industrial Internet of Things (IIoT) and associated computing continuum Cloud and Edge Technologies and Industrial AI (Artificial Intelligence) provide valuable data for Predictive Maintenance and product-oriented ZDM. Associated to that, ontologies and associated semantic technologies such as Knowledge Graphs are rapidly becoming popular in various domains and applications to deal with adding semantic meaning to this data and enable reasoning and queries.

All of the above is making the smart maintenance and manufacturing development with increasing "cognitive" and "predictive" characteristics to augment the human-machine collaboration.

In this Research Topic, we present a compilation of eight papers presenting and demonstrating results of recent research and innovation activity in a variety of topics within Data Driven Cognitive Manufacturing with applications in ZDM and Predictive Maintenance.

The first paper, "Product Quality Improvement Policies in Industry 4.0: Characteristics, Enabling Factors, Barriers, and Evolution Toward Zero Defect Manufacturing" by Foivos Psarommatis, Sylvain Prouvost, Gökan May and Dimitris Kiritsis presents a literature review on the implementation of these philosophies to improve quality of processes and products in a system, and also covers the commonalities and differences with Zero Defect Manufacturing (ZDM) philosophy. In this study, 45 articles have been analyzed. A categorization of quality improvement methods and the way toward ZDM is also presented and discussed.

The second paper, "Physically Inspired Data Compression and Management for Industrial Data Analytics" by Ramin Sabbagh, Zicheng Cai, Alec Stothert and Dragan Djurdjanovic describe a novel method that facilitates automated signal parsing into a set of exhaustive and mutually exclusive

OPEN ACCESS

Edited and reviewed by:
Kristof Van Laerhoven, University of Siegen, Germany

****Correspondence:***
Dimitris Kiritsis
dimitris.kiritsis@epfl.ch

Specialty section:
This article was submitted to Mobile and Ubiquitous Computing, a section of the journal Frontiers in Computer Science

Received: *26 November 2020*
Accepted: *21 December 2020*
Published: *08 February 2021*

Citation:
Kiritsis D, Lazaro O, Hodkiewicz M, Lee J and Ni J (2021) Editorial: Data-Driven Cognitive Manufacturing—Applications in Predictive Maintenance and Zero Defect Manufacturing. Front. Comput. Sci. 2:633850. doi: 10.3389/fcomp.2020.633850

segments, which is coupled with extraction of physically interpretable signatures that characterize those segments. The resulting numerical signatures can be used to approximate a wide range of signals within some arbitrary accuracy, thus effectively turning the aforementioned signal parsing and signature extraction procedure into a signal compression process. Application to multiple large datasets of sensor readings collected from several advanced manufacturing plants showed the feasibility of physics-inspired compression of industrial data.

The aim of the third paper, “Ontology-Based Context Modeling in Physical Asset Integrity Management” by Ali Al-Shdifat, Christos Emmanouilidis, Muhammad Khan and Andrew G. Starr is 2-fold: to analyse current approaches to addressing IoT context information management, mapping how context-aware computing addresses key challenges and supports the delivery of monitoring solutions; and to develop a maintenance context ontology focused on failure analysis of mechanical components so as to drive monitoring services adaptation. The approach is demonstrated by applying the ontology on an industrially relevant physical gearbox test rig.

The fourth paper, “Predictive Maintenance for Injection Molding Machines Enabled by Cognitive Analytics for Industry 4.0” by Vaia Rousopoulou, Alexandros Nizamis, Thanasis Vafeiadis, Dimosthenis Ioannidis and Dimitrios Tzovaras introduces a cognitive analytics, self- and autonomous-learned system bearing predictive maintenance solutions for Industry 4.0. A complete methodology for real-time anomaly detection on industrial data and its application on injection molding machines are presented.

The fifth paper, “Prognostics and Health Management of Industrial Assets: Current Progress and Road Ahead” by Luca Biggio and Iason Kastanis presents a thorough review of existing works both in the contexts of fault diagnosis and fault prognosis, highlighting the benefits and the drawbacks introduced by the adoption of AI techniques. The goal of the authors in this paper is to highlight potentially fruitful research directions along with characterizing the main challenges that need to be addressed in order to realize the promises of AI-based Prognostics and Health Management systems.

In the sixth paper, “Implementation and Transfer of Predictive Analytics for Smart Maintenance: A Case Study” by Sebastian Von Enzberg, Thanasis Naskos, Ifigeneia Metaxa, Daniel Köchling and Arno Kühn, the authors present a case study motivated by a typical maintenance activity in an industrial plant. The paper focuses on the crucial aspects of each phase of the Predictive Maintenance implementation process, towards the holistic integration of the solution within a company. A concept is derived for the model transfer to a different factory. This is illustrated by practical examples from a lighthouse factory within the BOOST 4.0 H2020 project.

The seventh paper, “Intelligent Predictive Maintenance and Remote Monitoring Framework for Industrial Equipment based on Mixed Reality” by Dimitris Mourtzis, John Angelopoulos and Nikos Panopoulos propose an approach for the modelling, design and development of a Predictive Maintenance and Remote Monitoring system, based on the utilization of AI algorithms for the data acquisition, fusion, and post-processing. In addition to that, the proposed framework will integrate a Mixed Reality application for the intuitive visualization of the data, that will ultimately facilitate production and maintenance engineers to monitor the condition of the machines and most importantly to get an accurate prediction of the oncoming failures.

The final paper, “RECLAIM: Towards new era of refurbishment and re-manufacturing of industrial equipment” by Angeliki Zacharaki, Thanasis Vafeiadis, Nikolaos Kolokas, Yuchun Xu, Michael Pesch, Dimosthenis Ioannidis and Dimitrios Tzovaras presents a new idea on refurbishment and re-manufacturing based on big data analytics, machine learning, predictive analytics and optimization models using deep learning techniques and digital twin models with the aim of facilitating the stakeholders to make informed decisions about whether to re-manufacture, upgrade or repair heavy machinery that is towards its end-of-life.

AUTHOR CONTRIBUTIONS

All authors listed have made a substantial, direct and intellectual contribution to the work, and approved it for publication.

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

REVIEW
published: 11 August 2020
doi: 10.3389/fcomp.2020.00026

Product Quality Improvement Policies in Industry 4.0: Characteristics, Enabling Factors, Barriers, and Evolution Toward Zero Defect Manufacturing

Foivos Psarommatis, Sylvain Prouvost, Gökan May and Dimitris Kiritsis*

ICT for Sustainable Manufacturing, École Polytechnique Fédérale de Lausanne, SCI-STI-DK, Lausanne, Switzerland

OPEN ACCESS

Edited by:
Duck Young Kim,
Pohang University of Science and Technology, South Korea

Reviewed by:
Jongho Shin,
Department of Industrial Engineering, Chosun University, South Korea
Hongbae Jun,
Hongik University, South Korea

****Correspondence:***
Foivos Psarommatis
foivos.psarommatis@epfl.ch

Specialty section:
This article was submitted to Mobile and Ubiquitous Computing, a section of the journal Frontiers in Computer Science

Received: *09 March 2020*
Accepted: *22 June 2020*
Published: *11 August 2020*

Citation:
Psarommatis F, Prouvost S, May G and Kiritsis D (2020) Product Quality Improvement Policies in Industry 4.0: Characteristics, Enabling Factors, Barriers, and Evolution Toward Zero Defect Manufacturing. Front. Comput. Sci. 2:26. doi: 10.3389/fcomp.2020.00026

In the competitive market of manufacturing, quality is a criterion of primary importance in order to win market share. Quality improvement must be coupled with performance point of view. Lean Manufacturing, Six Sigma, Lean Six Sigma, Total Quality Management, Theory of Constraints, and their combination are philosophies dedicated to this goal. This study is a literature review on the implementation of these philosophies to improve quality of processes and products in a system, and also covers the commonalities and differences with Zero Defect Manufacturing (ZDM) philosophy. In this study, 45 articles have been analyzed. These articles have been selected by a research on several scientific libraries with specific keywords. The methodology is based on a list of information extracted from each paper. The data searched are on the tool selections, critical factors of implementations and the benefits obtained from them. Based on the review and analysis of the literature and practices, we provide the top 10 main components of the tools used for quality improvement, enabling factors, benefits, and barriers to implementation. Moreover, we present and discuss categorization of quality improvement methods and the way toward ZDM. The need of standardized toolkits for different levels of maturity in quality management systems and a better education have been enlightened. Thanks to technological improvement in information flow management, ZDM seems close to be achieved even though some new risks and wastes have to be taken care of within the implementation.

Keywords: quality improvement philosophies, zero defect manufacturing, lean, six sigma, theory of constraints, total quality management, state of the art, review

INTRODUCTION

"If people were all the same we would not need to make so many kinds of printers, but people are different" (Yamashina, 1995). In the context of this globalized, ultra-connected world, benchmarking leads to a large number of competitive solutions to answer a need (Martins et al., 2015; Gillen, 2017). For a company, increasing and even keeping its market share is tougher than ever. One of the main factors that drives a product's commercial success is its quality (Wilson et al., 2016). The willingness to live of an organization then depends on strongly feeding research on

quality in order to provide to the customers a product that satisfies the most of their needs and even sublimates them. Nevertheless, a need is not defined in a fixed manner. It evolves and so does the manufacturing to produce the items. This evolution has organizations permanently questioning the quality of their products and processes, and binds them into a continuous improvement (CI) initiative to stay competitive (Singh and Singh, 2012; Kumar et al., 2018).

CI is done using Quality Management Systems (QMS) which rely on philosophies such as Lean Manufacturing (LM), Six Sigma (SS), Theory of Constraints (TOC), Total Quality Management (TQM), and Lean Six Sigma (L6S) (Hutchins, 2016). These philosophies are implemented through a large number of tools. The QMS efficiency may vary depending on some factors which can lead to failure of implementation (Nanda, 2005). It is important to understand these reasons in order to learn from the past and evolve positively (Cannon and Edmondson, 2005). Concerning quality improvement principles, several literature reviews have already been done in the past. Some new implementations are done every day and change is permanent (Rothwell et al., 2015). Moreover, thanks to technology improvement, Zero Defect Manufacturing (ZDM) is a philosophy for which the implementation is closer than ever (Eleftheriadis and Myklebust, 2016). This justifies a literature review of LM, SS, L6S, TOC, and TQM. The purpose of this review is to analyze the quality improvement tools used in manufacturing, the critical factors and benefits of implementation of these philosophies, and to investigate how they are related to ZDM. In addition to that, critical success and failure factors, and benefits of the five quality improvement philosophies, have been identified and ranked in order to get a better understanding of their impact.

The structure of the paper is as follows: section Research Method describes the methodology of the research, and section Previous Literature analyses the previous literature. Section 4 presents the results of the review by providing the Critical Success Factors (CSF), Critical Failure Factors (CFF), benefits, and analysis of quality improvement tools. Next, section Review of Results highlights the need for standardization and categorization of quality improvement tools, along with discussions on CSF, CFF, and benefits illustrating the evolution toward ZDM. Finally, section Discussion ends the paper by highlighting the main findings and outcomes of the study. For the ease of the reader a list with all the abbreviations used in the paper are summarized in **Table 1**.

RESEARCH METHOD

The purpose of this paper is to do a systematic review of the quality improvement tools used in the manufacturing domain. In order to acquire a representing sample of papers, the following steps have been followed. The first step was to create a search query for conducting the search. The query used can be seen below.

- TITLE(((Lean) OR (Six sigma OR 6S OR SS) OR ((Lean six sigma) OR LSS OR L6S) OR (TQM OR (total quality management)) OR (ToC OR (Theory of constrains))) AND (review OR (State of the art) OR (literature review)) AND (manufacturing OR production)).

This query was used in different scientific databases; more specifically, the search was done in Engineering Village (Compendex and Inspec), Scopus, Web of Science, and Science Direct. In total, 383 articles were found, after removing the duplicates. The next step was to filter them based on the relevance and if the full article was available. After this filtering, 45 articles have been selected to conduct the analysis.

The acronyms LM, SS, L6S, TOC and TQM stand for:

- **LM**: Lean Manufacturing is a philosophy oriented toward waste reduction. Seven wastes have been identified: overproduction or asynchrony (producing too much or in an inadequate timing), inventory (store raw material, work in process, and finished products), motion (unnecessary body movement), defectiveness (non-conforming product), transportation (unnecessary movement of product), overprocessing (processing beyond customer expectations), and waiting (time spent before next activity) (Chiarini, 2013).
- **SS**: Six Sigma is a statistical philosophy oriented to product or process variability reduction. The desired result is defined depending on the customer need and its vision of defect in order to ensure the customer satisfaction (Linderman et al., 2003).
- **L6S**: Lean Six Sigma is a combination of LM and SS. The idea is that SS focuses well on quality while LM focuses on the speeding process. Their combination helps to reach a state of statistical control and operational improvements (Atmaca and Girenes, 2011).

TABLE 1 | Abbreviation list.

Abbreviation	Description
5S	Sort, Set In order, Shine, Standardize and Sustain
CFF	Critical Failure Factors
CI	Continuous Improvement
CSF	Critical Success Factors
DMAIC	Define, Measure, Analyse, Improve, and Control
FMEA	Failure Mode and Effects Analysis
JIT	Just in Time
L6S	Lean Six Sigma
LM	Lean Manufacturing
QMS	Quality Management Systems
SME	Small Medium Enterprises
SPC	Statistical Process Control
SS	Six Sigma
TOC	Theory of Constrains
TPM	Total Productive Maintenance
TQM	Total Quality Management
VSM	Value Stream Mapping
ZDM	Zero Defect Manufacturing

- **TOC**: Theory of Constraints is a philosophy is based on the idea that a manufacturing system is constrained. Its quality can be measured by throughput, inventory, and operational expenses. The goal is to maximize throughput while decreasing the inventory and operational expenses. This is done by identifying the constraints, deciding how to exploit them, aligning the system on the exploitation decision, elevating the system's constraints, and by iterating, if during the process, one of the constraints has been broken (Goldratt, 2020).
- **TQM**: Total Quality Management is a philosophy focused on the organization's culture of quality. It is mainly a mindset that everyone in an organization must be dedicated to give its best in order to provide high standards quality on the result of activities done. The goal is to reduce errors, improve customer and employee experience (Martínez-Lorente et al., 1998).

PREVIOUS LITERATURE

The selected articles are literature review articles. As manufacturing and quality management evolve over time, the findings and conclusions of literature reviews may differ depending on when the review had been carried out. Therefore, it is important to take the time to analyze this evolution by reviewing these previous works. They have been categorized depending on the quality philosophy they focus on. The classification takes into account the main philosophy studied and if it is studied combined with other philosophies. Only one paper (Kedar et al., 2008) studied a combination of the five philosophies. Seven papers (Arnheiter and Maleyeff, 2005; Bendell, 2006; Kasemset, 2011; Alhuraish et al., 2015; Pacheco et al., 2015; Costa et al., 2018; Stankalla et al., 2018; Makwana and Patange, 2019) have focused on a combination of two philosophies and the rest has concentrated the effort on one main one. **Figure 1** illustrates the distribution by year of the analyzed review articles. The results show that there is an increasing trend after 2014 for review articles regarding the five philosophies. Further to that, **Figure 1** shows the corresponding percentage of each philosophy; the highest percentage corresponds to SS (30.19%), and then the LM and L6S follow with 24.53 and 26.42%, respectively. The philosophy with the least number of review papers is the TOC, mostly because it has already been proved to be inefficient to solve multi-constraints problems (Rahman, 1998; Ikeziri et al., 2019).

While **Figure 1** shows that there is a higher interest for the studying philosophies the past 6 years, the results does not capture in detail the trends, because the articles used for **Figure 1** were review papers. In this regard, the references of those 45 analyzed papers were considered and classified based on their year. In total, the 45 analyzed papers had 4,125 references that they were considering. The results from the year analysis of those 4,125 articles are shown in **Figure 2**. The period that those references were covering was from 1970 to 2019. From 1970 to 2004, all the philosophies show an increasing trend, with the

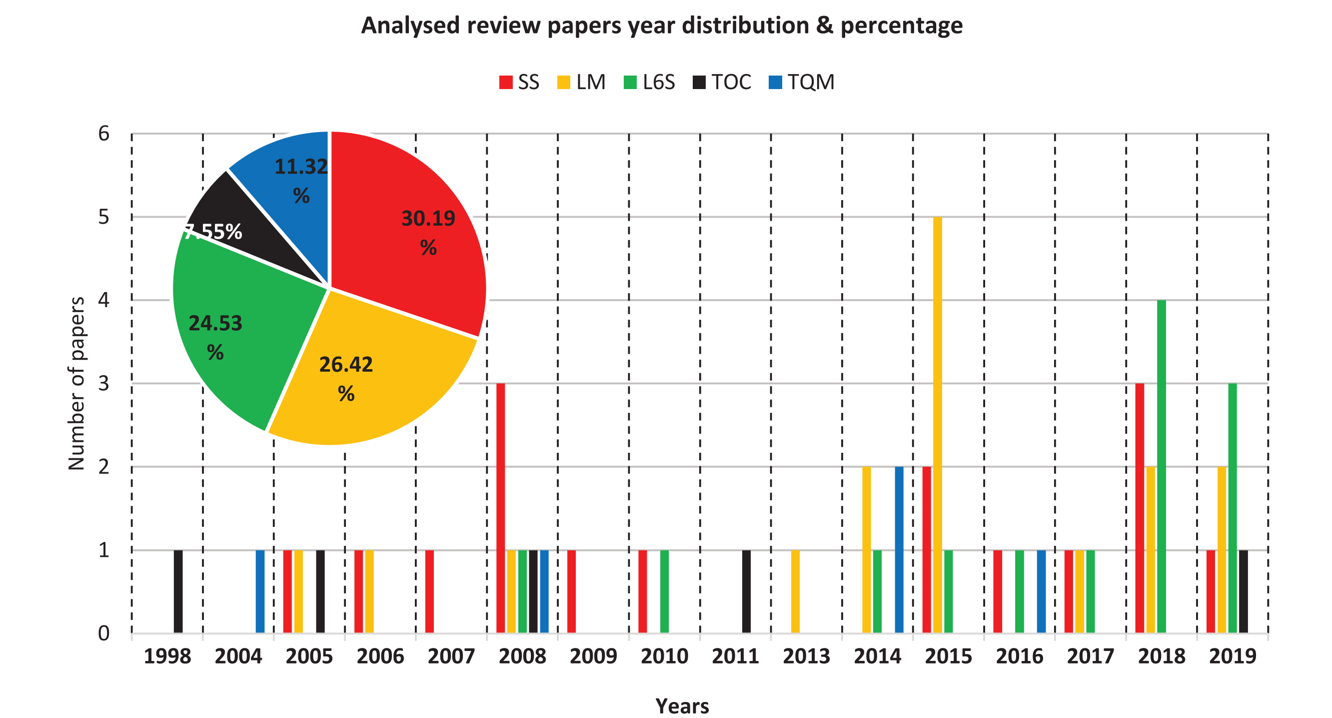


FIGURE 1 | Distribution of the philosophy focus regarding the literature.

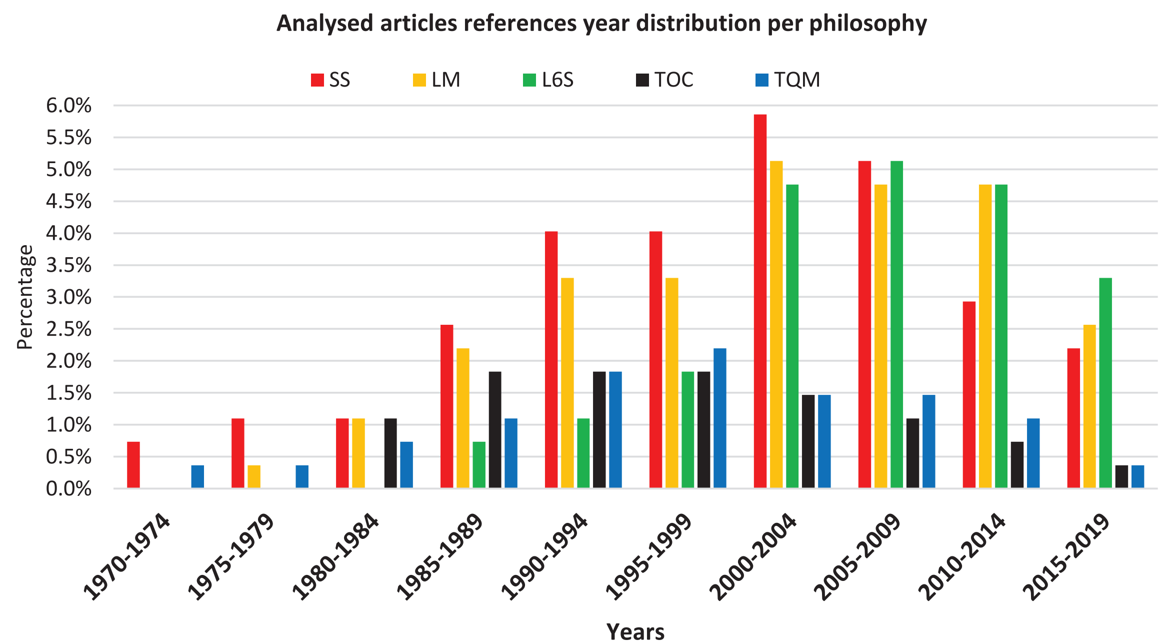


FIGURE 2 | Distribution of the references mentioned in the reviewed papers through time.

most popular being the SS approach, followed by LM. After 2004 there is a decreasing trend, and also SS stops being the top category; its place is taken by L6Sm followed by LM. TOC and TQM have the smallest percentages, but they show similar trend as the leading philosophes; they have an increasing trend until 1999, and afterwards, they decrease to a point that becoming significantly less than the other three methods. Lately, the main focus of the research done is to understand which tool can be used for which purpose and how to standardize them. On one hand, TQM is more a philosophy and a mindset to spread in the organization than a toolbox, and hence, less studies investigate this topic. On the other hand, the philosophies LM, SS, and their more recent combination, L6S, have seen their toolbox being more and more furnished thanks to technology, for example (Gladysz and Buczacki, 2018).

Finally, the decreasing trend on the last year is also due to some inertia. Indeed, the time data presented in **Figure 2** presents the range of publication years of the references of the 45 reviewed papers analyzed. Therefore, inevitably there is a delay between the publication year of an article and its use in other articles, but it captures the current trend.

REVIEW OF RESULTS

This section deals with an analysis of the current philosophies (LM, SS, L6S, TOC, and TQM) used for quality improvement management. Of the 45 articles reviewed, CSF, CFF, and benefits have been pulled out in order to get an understanding for some causes in the success of an implementation of the different philosophies. For these three categories, the top 10 are listed in **Tables 2–4**. The ranking has been established depending on the number of different articles citing the notion. The more a notion is mentioned, the higher the ranking is. For example, in **Table 2** for the CSF "*Trained workers on the philosophies of quality improvement*," there are nine different articles mentioning it. Its weight is then of 9. This is more than "*A good identification and prioritization of improvements to do*" with a weight of 8 and less than "*A visible Top Management commitment*" with 11. Therefore, they are ranked in the order as second, third, and fourth CSF. In this analysis, the three studied notions are defined as:

- **CSF**, as a strong cause that has helped in previous successful implementations of quality improvement philosophy.
- **CFF**, as a strong cause that has hampered successful implementations of quality improvement philosophy.
- **Benefit**, as a positive consequence from successful implementations of quality improvement philosophy.

Enabling Factors and Benefits

Based on the review and analysis of the literature and practices, the enabling factors for quality improvement in the form of Top 10 CSFs are presented below:

- **Use of proper communication to avoid misunderstanding:** As explained in the first point, strategies of quality improvement are defined by the top management. Afterwards, the message is carried by intermediates across the

TABLE 2 | Top 10 of CSF identified in the articles reviewed.

1	**CSF**	**Use of a proper communication to avoid misunderstanding**
	LM	(Gupta and Jain, 2013; Bhamu and Singh Sangwan, 2014; Sundar et al., 2014; Yusup et al., 2015)
	SS	(Oke, 2007; Kumar and Antony, 2008; Gamal Aboelmaged, 2010; Stankalla et al., 2018)
	L6S	(Lande et al., 2016; Antony et al., 2017; Muraliraj et al., 2018; Ruben et al., 2018; Stankalla et al., 2018)
2	**CSF**	**Visible management commitment**
	LM	(Costa et al., 2018)
	SS	(Oke, 2007; Kumar and Antony, 2008; Gamal Aboelmaged, 2010; Tjahjono et al., 2010; Costa et al., 2018; Stankalla et al., 2018)
	L6S	(Lande et al., 2016; Antony et al., 2017; Muraliraj et al., 2018; Ruben et al., 2018; Stankalla et al., 2018)
	TQM	(Al-Khalili and Subari, 2014)
3	**CSF**	**Trained workers on the philosophies of quality improvement**
	LM	(Gupta and Jain, 2013; Bhamu and Singh Sangwan, 2014; Sundar et al., 2014; Costa et al., 2018)
	SS	(Oke, 2007; Kumar and Antony, 2008; Gamal Aboelmaged, 2010; Costa et al., 2018; Stankalla et al., 2018)
	L6S	(Lande et al., 2016; Muraliraj et al., 2018; Ruben et al., 2018; Stankalla et al., 2018)
4	**CSF**	**Good identification and prioritization of improvements**
	LM	(Alhuraish et al., 2015)
	SS	(Kumar and Antony, 2008; Gamal Aboelmaged, 2010; Alhuraish et al., 2015; Alcaide-Muñoz and Gutierrez-Gutierrez, 2017; Stankalla et al., 2018)
	L6S	(Lande et al., 2016; Alsaffar and Ketan, 2018; Muraliraj et al., 2018; Stankalla et al., 2018)
	TOC	(Kirche and Srivastava, 2005)
5	**CSF**	**Strong link between philosophy, business strategy, and customer satisfaction**
	SS	(Kumar and Antony, 2008; Gamal Aboelmaged, 2010; Tjahjono et al., 2010; Stankalla et al., 2018)
	L6S	(Lande et al., 2016; Antony et al., 2017; Muraliraj et al., 2018; Ruben et al., 2018; Stankalla et al., 2018)
6	**CSF**	**Good understanding of tool choice depending on the goal**
	LM	(Gupta and Jain, 2013)
	SS	(Tjahjono et al., 2010; Stankalla et al., 2018)
	L6S	(Ruben et al., 2018; Stankalla et al., 2018)
	TOC	(Kasemset, 2011)
	TQM	(Al-Khalili and Subari, 2014)
7	**CSF**	**Use of precise quantification tools**
	LM	(Bendell, 2006; Alhuraish et al., 2015; Pacheco et al., 2015)
	SS	(Bendell, 2006; Gamal Aboelmaged, 2010; Alhuraish et al., 2015; Pacheco et al., 2015; Alcaide-Muñoz and Gutierrez-Gutierrez, 2017)
	L6S	(Lande et al., 2016)
8	**CSF**	**Linking QMS to the global supply chain**
	LM	(Bhamu and Singh Sangwan, 2014)
	SS	(Kumar and Antony, 2008; Tjahjono et al., 2010; Stankalla et al., 2018)
	L6S	(Lande et al., 2016; Muraliraj et al., 2018; Stankalla et al., 2018)
9	**CSF**	**Systemic approach to improve by iterations**
	LM	(Bhamu and Singh Sangwan, 2014; Sundar et al., 2014)
	TOC	(Rahman, 1998; Kasemset, 2011; Ikeziri et al., 2019)
10	**CSF**	**Strong involvement of employees**
	LM	(Bhamu and Singh Sangwan, 2014; Sundar et al., 2014)
	SS	(Kumar and Antony, 2008)
	L6S	(Lande et al., 2016; Ruben et al., 2018)

organization. Proper communication helps to avoid distortion of the message or even distortion of the need.

- **Visible management commitment:** Quality strategies are determined by the top management of the organization. These guidelines have to be broadcast across the whole structure up to the shop floor. The more intermediates (like managers) are used to carry the message—the more distant the decision makers seem, the less concerned the employees may be. Therefore, top management must shorten this distance by being committed to these guidelines and showing the employees they care about it.
- **Trained actors on the philosophies of quality improvement:** Trained workers are used to implement new systems. They understand well the causes and consequences of choices in terms of impact on the system, the products, the employees, and more, in order to do their best to succeed.

TABLE 3 | Top 10 benefits (BFT) identified in the articles reviewed.

1	**BFT**	**Cost reduction**
	LM	(Bendell, 2006; Gupta and Jain, 2013; Nithia et al., 2015; Pacheco et al., 2015; Costa et al., 2018; Ismail et al., 2019)
	SS	(Bendell, 2006; Oke, 2007; Kumar and Antony, 2008; Gamal Aboelmaged, 2010; Pacheco et al., 2015; Costa et al., 2018; Patel and Desai, 2018)
	L6S	(Alsmadi and Khan, 2010; Antony et al., 2017; Alexander et al., 2019; Siregar et al., 2019)
	TOC	(Kasemset, 2011)
2	**BFT**	**Lead-time reduction**
	LM	(Gupta and Jain, 2013; Bhamu and Singh Sangwan, 2014; Pinho and Mendes, 2017; Costa et al., 2018; Gladysz and Buczacki, 2018)
	SS	(Oke, 2007; Kumar and Antony, 2008; Costa et al., 2018; Patel and Desai, 2018)
	L6S	(Alsmadi and Khan, 2010; Antony et al., 2017; Alexander et al., 2019)
	TOC	(Rahman, 1998; Ikeziri et al., 2019)
	TQM	(Nandurkar et al., 2014)
3	**BFT**	**Quality improvement**
	LM	(Gupta and Jain, 2013; Bhamu and Singh Sangwan, 2014; Yusup et al., 2015; Pinho and Mendes, 2017; Costa et al., 2018; Ismail et al., 2019)
	SS	(Oke, 2007; Gamal Aboelmaged, 2010; Alcaide-Muñoz and Gutierrez-Gutierrez, 2017; Costa et al., 2018)
	L6S	(Albliwi et al., 2015; Antony et al., 2017; Alexander et al., 2019; Siregar et al., 2019)
4	**BFT**	**Inventory reduction**
	LM	(Gupta and Jain, 2013; Sundar et al., 2014; Pinho and Mendes, 2017; Gladysz and Buczacki, 2018)
	SS	(Oke, 2007; Gamal Aboelmaged, 2010)
	L6S	(Alsmadi and Khan, 2010; Albliwi et al., 2015; Antony et al., 2017; Siregar et al., 2019)
	TOC	(Rahman, 1998; Ikeziri et al., 2019)
	TQM	(Nandurkar et al., 2014)
5	**BFT**	**Cycle time reduction**
	LM	(Gupta and Jain, 2013; Bhamu and Singh Sangwan, 2014; Nithia et al., 2015; Yusup et al., 2015)
	SS	(Oke, 2007; Kumar and Antony, 2008; Gamal Aboelmaged, 2010)
	L6S	(Albliwi et al., 2015; Antony et al., 2017; Alsaffar and Ketan, 2018; Siregar et al., 2019)
6	**BFT**	**Increased production**
	LM	(Gupta and Jain, 2013; Bhamu and Singh Sangwan, 2014; Costa et al., 2018)
	SS	(Oke, 2007; Kumar and Antony, 2008; Gamal Aboelmaged, 2010; Costa et al., 2018)
	L6S	(Alsmadi and Khan, 2010; Albliwi et al., 2015; Alsaffar and Ketan, 2018; Alexander et al., 2019; Siregar et al., 2019)
	TQM	(Nandurkar et al., 2014)
7	**BFT**	**Increased customer satisfaction**
	LM	(Bendell, 2006; Kedar et al., 2008; Bhamu and Singh Sangwan, 2014)
	SS	(Bendell, 2006; Kedar et al., 2008; Gamal Aboelmaged, 2010; Patel and Desai, 2018)
	L6S	(Albliwi et al., 2015; Antony et al., 2017; Siregar et al., 2019)
	TQM	(Al-Khalili and Subari, 2014)
8	**BFT**	**Reduction of variability in quality**
	LM	(Kedar et al., 2008; Bhamu and Singh Sangwan, 2014; Sundar et al., 2014; Alhuraish et al., 2015)
	SS	(Kedar et al., 2008; Gamal Aboelmaged, 2010; Alhuraish et al., 2015; Alcaide-Muñoz and Gutierrez-Gutierrez, 2017)
	L6S	(Kedar et al., 2008; Alsmadi and Khan, 2010; Albliwi et al., 2015; Siregar et al., 2019)
9	**BFT**	**Employee morale improvement**
	LM	(Gupta and Jain, 2013; Sundar et al., 2014; Pacheco et al., 2015; Pinho and Mendes, 2017)
	SS	(Kumar and Antony, 2008; Pacheco et al., 2015)
	L6S	(Alsaffar and Ketan, 2018; Alexander et al., 2019; Siregar et al., 2019)
10	**BFT**	**On-time delivery increase**
	LM	(Bhamu and Singh Sangwan, 2014; Pacheco et al., 2015; Pinho and Mendes, 2017; Ismail et al., 2019)
	SS	(Oke, 2007; Gamal Aboelmaged, 2010; Pacheco et al., 2015; Patel and Desai, 2018)
	TOC	(Rahman, 1998; Ikeziri et al., 2019)

- **Good identification and prioritization of improvements:** An organization may have several domains of improvement. First of all, it is important to identify all opportunities. This is of primary importance, as some improvements might have precedence over another. Moreover, some improvements are more critical than others. Therefore, one must be able to identify the full list of improvements and determine the right timing for each one to be done.

TABLE 4 | Top 10 of CFF identified in the articles reviewed.

1	**CFF**	**Lack of implementation experience and training for actors**
	LM	(Arnheiter and Maleyeff, 2005; Gupta and Jain, 2013; Bhamu and Singh Sangwan, 2014; Sundar et al., 2014; Nithia et al., 2015; Yusup et al., 2015; Pinho and Mendes, 2017; Costa et al., 2018)
	SS	(Arnheiter and Maleyeff, 2005; Oke, 2007; Kumar and Antony, 2008; Van Iwaarden et al., 2008; Tjahjono et al., 2010; Costa et al., 2018)
	L6S	(Albliwi et al., 2015; Antony et al., 2017; Muraliraj et al., 2018; Ruben et al., 2018; Alexander et al., 2019; Siregar et al., 2019)
	TQM	(Al-Khalili and Subari, 2014; Dedy et al., 2016)
2	**CFF**	**Lack of top management commitment**
	LM	(Gupta and Jain, 2013; Bhamu and Singh Sangwan, 2014; Alhuraish et al., 2015; Nithia et al., 2015)
	SS	(Kumar and Antony, 2008; Van Iwaarden et al., 2008; Tjahjono et al., 2010; Alhuraish et al., 2015)
	L6S	(Alsmadi and Khan, 2010; Albliwi et al., 2015; Muraliraj et al., 2018; Ruben et al., 2018; Alexander et al., 2019; Siregar et al., 2019)
	TQM	(Al-Khalili and Subari, 2014; Dedy et al., 2016)
3	**CFF**	**Resistance to change**
	LM	(Gupta and Jain, 2013; Alhuraish et al., 2015; Nithia et al., 2015; Costa et al., 2018)
	SS	(Kumar and Antony, 2008; Van Iwaarden et al., 2008; Tjahjono et al., 2010; Alhuraish et al., 2015; Costa et al., 2018)
	L6S	(Alsmadi and Khan, 2010; Albliwi et al., 2015; Antony et al., 2017; Muraliraj et al., 2018; Ruben et al., 2018; Alexander et al., 2019)
4	**CFF**	**Lack of resources**
	LM	(Gupta and Jain, 2013; Alhuraish et al., 2015; Nithia et al., 2015; Pinho and Mendes, 2017)
	SS	(Kumar and Antony, 2008; Tjahjono et al., 2010; Alhuraish et al., 2015; Stankalla et al., 2018)
	L6S	(Albliwi et al., 2015; Antony et al., 2017; Muraliraj et al., 2018; Ruben et al., 2018; Stankalla et al., 2018; Alexander et al., 2019)
5	**CFF**	**Lack of employee involvement**
	LM	(Bendell, 2006; Yusup et al., 2015)
	SS	(Bendell, 2006; Oke, 2007; Kumar and Antony, 2008; Van Iwaarden et al., 2008)
	L6S	(Alsmadi and Khan, 2010; Muraliraj et al., 2018; Ruben et al., 2018; Siregar et al., 2019)
6	**CFF**	**Lack of framework of implementation**
	LM	(Bhamu and Singh Sangwan, 2014; Pacheco et al., 2015; Pinho and Mendes, 2017)
	SS	(Oke, 2007; Gamal Aboelmaged, 2010; Tjahjono et al., 2010; Pacheco et al., 2015; Alcaide-Muñoz and Gutierrez-Gutierrez, 2017; Patel and Desai, 2018)
	L6S	(Antony et al., 2017)
7	**CFF**	**Need for a specialist**
	LM	(Bendell, 2006; Kedar et al., 2008)
	SS	(Bendell, 2006; Oke, 2007; Kedar et al., 2008; Gamal Aboelmaged, 2010)
	L6S	(Kedar et al., 2008; Antony et al., 2017; Muraliraj et al., 2018; Alexander et al., 2019; Siregar et al., 2019)
	TOC	(Kedar et al., 2008)
	TQM	(Talha, 2004; Kedar et al., 2008)
8	**CFF**	**Poor communication system**
	LM	(Bendell, 2006; Nithia et al., 2015; Yusup et al., 2015)
	SS	(Bendell, 2006)
	L6S	(Albliwi et al., 2015; Ruben et al., 2018; Siregar et al., 2019)
	TQM	(Dedy et al., 2016)
9	**CFF**	**Data control infrastructure implementation difficulties**
	LM	(Kedar et al., 2008; Bhamu and Singh Sangwan, 2014; Alhuraish et al., 2015)
	SS	(Kedar et al., 2008; Gamal Aboelmaged, 2010; Alhuraish et al., 2015)
	L6S	(Kedar et al., 2008; Albliwi et al., 2015; Siregar et al., 2019)
10	**CFF**	**Poor selection of projects**
	SS	(Kumar and Antony, 2008; Tjahjono et al., 2010)
	L6S	(Albliwi et al., 2015; Antony et al., 2017; Muraliraj et al., 2018; Alexander et al., 2019)

- **Strong link between philosophy, business strategy, and customer satisfaction:** The main goal of a manufacturing system is to bring products that will fulfill one or more customer needs. Therefore, strategic manufacturing choices must be linked to the customer need in order to ensure their satisfaction. Once the choices are made, the changes induced must be treated with the adequate philosophy leading to the greatest customer satisfaction.
- **Good understanding of tool choice depending on the goal:** Several philosophies exist to implement quality

improvements, as well as several goals for quality improvement. Therefore, tools may be useful to reach some targets, but not all. A good knowledge of which tool is useful for which purpose is important in order to use them in the appropriate manner.

- **Use of precise quantification tools:** Improvement is defined as a positive change from an initial state to a final one. These states must be well-determined to precisely estimate the gain. Therefore, tools must be able to accurately quantify the chosen units for improvement.
- **Linking QMS to the global supply chain:** The manufacturing department strongly relies on suppliers, logistic department, marketing department, etc. Therefore, QMS implementation will face some barriers which can be overcome by linking the actors of the supply chain to the QMS.
- **Systemic approach to improve by iterations:** An improvement is a change, and that means to move to a less known situation, as compared to before. A systemic approach helps to reduce the unknown during the implementation by applying a global method and avoiding particularities. By reducing the complexity and the impact of singularities on the system, a systemic approach helps to progress by iterations following a framework determined in advance.
- **Strong involvement of employees:** The different philosophies of quality improvement require that everyone is dedicated to providing the best quality possible. This means that ensuring the commitment of all the employees toward quality improvement is important to successfully evolve.

Based on the analysis performed on the selected articles, the Top 10 benefits as a result of the quality improvement policies are listed below. Furthermore, **Table 3** presents those benefits as classified per method, and provides the corresponding references.

Barriers

Some of the CSF previously presented can also turn out to be weaknesses if they are badly used. Besides, some important aspects must be taken into account to help avoid some failures. These barriers to the improvement of quality are explained in the form of Top 10 CFFs as follows:

- **Lack of implementation experience and training for actors:** In order to implement a QMS, people must have a certain knowledge of the philosophy and tools to avoid failures and useless expenses.
- **Lack of top management commitment:** The implementation of quality improvement systems needs the dedication of everyone in the organization. The will to increase quality often comes from the top management, which has a wider perspective on the product and the customer satisfaction. The top management then pushes the quality initiatives. In order to be credible, they must stay committed to this position to ensure a sustainable development.
- **Resistance to change:** Change means going from a well-understood state to another less known one. The actors of the organizations have to make an effort then to change. Some inertia may occur before benefits appear and then the will to go back to the previous state might pressure the organization. This is a resistance to change.
- **Lack of resources:** In order to change and improve, resources (human, financial, infrastructure) are used. Sometimes the change requires more resources than affordable for example in small and medium enterprises (SMEs).
- **Lack of employees' involvement:** As explained before, a strong involvement of employees is a strength whereas a weak one is a drawback.
- **Lack of framework of implementation:** The process of quality improvement is iterative and relies on several levels of maturity. There is no standard framework or procedure to follow. Thus, improvement is hardly reproducible.
- **Need for a specialist:** There is considerable knowledge in the quality management field. A CSF is to have a trained population. Therefore, there is the need of a specialist to accomplish this mission and have an expertise. This is an additional resource for the implementation of QMS.
- **Poor communication system:** As explained before, proper communication is important to ensure that actors are on the same page, and that results are provided and dynamically improve.
- **Data control infrastructure implementation difficulties:** Once the improvement is implemented, there is a phase of monitoring to measure the impact of the change. The measurements need an adequate data control infrastructure. Otherwise, the conclusion on benefits of the change is less precise.
- **Poor selection of projects:** In order to be sustainable, a strategic plan of improvements has to be designed. A poor selection and sequencing of projects may lead to failures or not fixing the main problems.

Analysis of Quality Improvement Tools

An analysis of the tools listed in the articles reviewed has been done. It is important to define what is considered to be a tool. First of all, a mechanism is considered to be a tool when the action of applying this mechanism enhances the quality of the product or the process. The top 10 tools are described in **Table 5**. The ranking is based on the weight attributed to each tool. The weight is determined as the number of different articles mentioning the tool all philosophies combined. In total, 144 tools have been listed. The focus of the study is on the tools which are mentioned in at least two different articles, which results in the further analysis of 99 tools. The total number of citations is 586 for these 99 tools, and each tool may be listed in several philosophies.

Furthermore, the tools have been clustered in two classes, based on the nature of the approach and the targeted goal. The first classification regroups the tools depending on their focus. The results from this classification are shown in **Table 6** and **Figure 3**. **Table 6** contains the overall results for all the philosophies with the corresponding percentages occurred from the analysis. There are five main categories regarding the focus of the tools, i.e., global improvement with 34.81% is the dominant category, followed by flow mapping and quality with 19.80 and 16.89%, respectively. The least developed tool is on the

TABLE 5 | Top 10 of listed tools in the literature review.

Rank	Tool	Definition	%
1	**DMAIC (**Define, Measure, Analyze, Improve, and Control**)**	DMAIC is a strategy of improvement originated from SS, but can be extended to LM and L6S.	4.09%
2	**VSM (**Value Stream Mapping**)**	VSM is a lean flow mapping tool. It is a process-oriented tool which map the value creation of the product, the time, resources spent, and information generated.	3.75%
3	**5S (**Sort, Set In Order, Shine, Standardize, and Sustain**)**	5S is an iterative lean tool which helps to sort, order, clean, and standardize a workplace in order to make it more efficient and improve the work life of the workplace's user. 5S stands for the initials of 5 Japanese words.	3.41%
4	**JIT (**Just In Time**)**	JIT is a lean flow timing management tool. Its goal is to align the timings in order the piece parts arrive just when needed for them to reach the next step of manufacturing for the product. The same reasoning stands for the finished good which arrive when needed. The purpose is to reduce the inventory of parts and finished goods and then the waiting time.	3.41%
5	**Kaizen**	Kaizen is a lean general approach for improvement in a system in order to reduce wastes.	2.90%
6	**Standardization**	Standardization is a tool which helps to reduce the variability of a process and product to ensure a consistent quality	2.90%
7	**TPM (**Total Productive Maintenance**)**	TPM is a tool designed to program the maintenance of the machines. The goal is to reduce the downtime and the unexpected stops.	2.90%
8	**Kanban**	Kanban is a lean tool to reach a pull system of quality. This relies on a visual tracking system of the product progress to well manage the flow.	2.73%
9	**SPC (**Statistical Process Control**)**	SPC is a statistical tool to quantify the variability of a process and monitor the current state of the situation.	2.55%
10	**FMEA (**Failure Mode and Effects Analysis**)**	FMEA is a tool listing the potential failures of a system. It lists and ranks the risks on human, methods and utilization, security and environmental factors to help tackle them in a cost and impact limited way.	2.38%

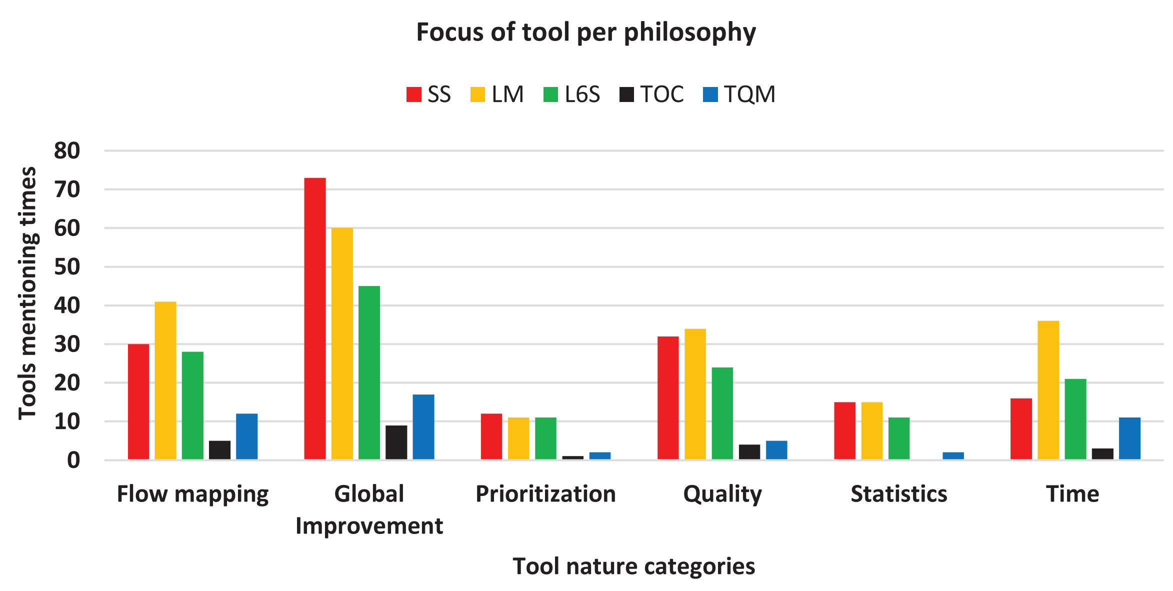


FIGURE 3 | Distribution of the mentions for the focus of the tools for each philosophy.

prioritization of the improvement to make with 6.31% of the category importance.

Figure 3 shows that most of the tools used in SS approach are used for global improvement. In addition to that, prioritization and statistics tools are almost equally used by SS, LM, and L6S approaches. Flow mapping and time tools are used more for LM, followed by SS and L6S, with similar mentions accordingly.

The second classification splits the tools depending on their goal. The results are shown in **Table 7** and **Figure 4**. In **Table 7**, there are two main goals that are dominant, i.e., the goal of determining current state of a system and

TABLE 6 | Definition of the different options of the first classification of tools on their nature.

Focus of tool	Definition	Percentage
Flow mapping	Tools that focus on modeling the succession of processes and identify value added (or not) steps.	19.80%
Global improvement	The tools that focus on several of the previous aspects and combine them to improve the quality of a system.	34.81%
Prioritization	Tools that focus on the prioritization of improvements, in order to efficiently improve the quality output of a system.	6.31%
Quality	Tools that focus on the quality of the product or process and how to avoid mistakes.	16.89%
Statistics	Tools that focus on statistics for a product (finite or not) at a fixed step of progress.	7.34%
Time	Tools that focus on time spent at each step of the manufacturing.	14.85%

TABLE 7 | Definition of the different options of the second classification of tools on their goal.

Tool goal	Definition	Percentage
All	The tool focus combines all the aspects of the previous impacts.	11.95%
Determining current state	The tool focus is on clearly determine the current situation of the system.	31.57%
Prevention	The tool focus is on preventing identified undesired quality scenarios to happen.	30.89%
Rethinking	The tool focus is on redesigning an element of the system. It has not necessarily presented problems, but quality improvements can be done to reach a better system.	9.90%
Solving	The tool focus is on solving a quality problem, which has already occurred.	15.70%

the goal of preventing problems from occurring, at 31.57 and 30.89% accordingly. The rethink goal is the least used among others with only 9.90%. **Figure 4** illustrates the goals of the identified tools per philosophy. In the two dominant categories, "Determining current state" and "Prevention," the philosophies with the highest use are SS and LM, followed by L6S. Furthermore, SS is the philosophy that uses all four goals the most. TQM shows a steadier presence in all five categories.

DISCUSSION

This section discusses the results and findings of the study. First, in section Need for Standardization of Quality Improvement Tools, we illustrate that some work on the standardization of the tool and the establishment of different toolkits to use depending on the level of maturity of the quality management system should be done. Subsequently, section Categorization of the Quality Improvement Tools provides two categorizations that help understand the way QMS are implemented and elaborates further on these categories and their implications. Next, section Discussions on CSF, CFF, and Benefits discusses and provides insights into the CSF, CFF, and benefits, previously highlighted in section Review of Results. Section Evolution Toward ZDM ends the discussion by linking the findings from the review on LM, SS, L6S, TOC, and TQM with some important factors and new possibilities of ZDM.

Need for Standardization of Quality Improvement Tools

The latest philosophy integrated in QMS is L6S, for which interest has risen since 2003 (Albliwi et al., 2015). After almost 20 years of study, more than the other four philosophies discussed in this article, many tools have been developed to help one implement a QMS. Through the literature review conducted, 99 tools have been listed as cited in more than one of the articles included in the scope. Several points can be brought to light from this listing.

Indeed, dealing with this amount and variety can feel like a barrier to increasing quality for those who are not specialists in their field. It means that first, the practitioner would have to spend a considerable amount of time to understand the tools, how they relate to each philosophy, and the critical points to ensuring the viability of the implementation of these tools. The manager can then decide either to do the work himself or to call for a specialist. In the end, both of these options represent a considerable consumption of resources (time or financial resources).

Moreover, different names of tools may refer to the same one. For example, some tools have been designed in Japan. For the sake of easier understanding and discussion, an English translation has been accepted by the community. Nevertheless, the translation is not standardized and several are accepted. A good example is about the tool *Poka-Yoke*. The goal of this tool is to add a visual alert if an error is made by an actor during the manufacturing. For example, imagine a production with batch of 10 items. On the shop floor, someone's mission is to create this batch. If he misses one item and puts 9 items in the cart for the batch, then an alert would appear to mention that the number of items is incorrect. In English, the accepted translations are *fool-proofing* and *mistake proofing*. Other examples exist, like *Heijunka*, which is *production leveling*; *Jidoka*, which is *process autonomation*; or *Ishikawa*, which is *fishbone diagram*. A more extended list of examples is in **Table 8**. These numerous names referring to same thing could be confusing when used. Indeed, in a team working on project, some may know one name and some another one. Training and educating the population is hard enough to avoid this confusion.

Education for these tools also bears discussion and adds to this confusion. Indeed, as pointed out by Gamal Aboelmaged (2010), there is growing student interest on the topic. Three problems have appeared. First, courses and the way of thinking have to

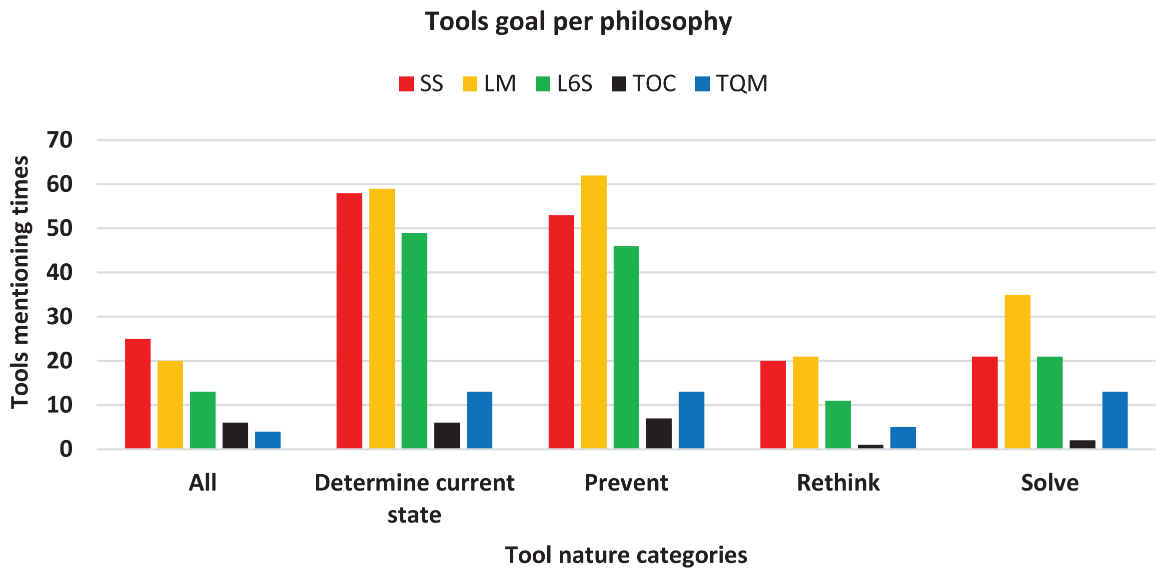


FIGURE 4 | Distribution of the mentions for the goal of the tools for each philosophy.

TABLE 8 | Tools synonyms in Japanese and English.

Japanese name	English name
Kaizen	Improvement project
Poka-Yoke	Mistake proofing/Fool proofing
Gemba walk	Go to the field
Jidoka	Process autonomation
Andon	Emergency stop for root-cause analysis
Heijunka	Production leveling
Hoshin	Policy deployment
Ishikawa	Fishbone diagram/Cause-and-effect diagrams
Taguchi	Design of experiments

be integrated to the spine of the education system. The second problem is inherent to the certifications provided by external institutions. Several institutions give lectures and certification known as *belts*. There is no standardization of them, as explained by Albliwi et al. (2015). Last but not least are issues explained by Edgeman and Dugan (2008). Edgeman and Dugan (2008) explain that SS is more than an engineering aspect. It is a philosophy which encompasses also human, societal, and environmental considerations. Therefore, there is a lack of harmonization for the content provided to become a specialist, whose expertise level can vary from one to another.

To conclude this discussion section, some work on the standardization of the tool and the establishment of different toolkits depending on the level of maturity of the QMS should be done.

Categorization of the Quality Improvement Tools

The two categorizations presented in **Figures 3**, **4** help to understand the way QMS are implemented. It is possible to reach to three conclusions. When studying these figures, it is useful to keep in mind how the construction was made. Tools have been cited in the articles reviewed. These articles were categorized depending on the main philosophies they focus on. Tools have also been categorized depending on the main goal they fulfill and the nature of their impact. When a tool is cited in an article for a philosophy, an increment of +1 is added on the categories of the corresponding main goal and nature for the conforming philosophy. One tool may appear in articles for different philosophies. Therefore, one tool can be found in only one category, but in several philosophies. If the same tool is cited several times in different articles of a same philosophy, then this tool will add an increment as big as the number of different articles on the same philosophy citing it. Tools that are cited in only one article have been removed from the focus of the study. A small column does not mean that none or few tools exist, but rather that they are less cited and then less studied. For example, LM tools are the most studied ones.

First of all, regarding the goals targeted, it is clearly visible that the dominant ones are about determining the current state of the organizations or preventing undesirable situations from occuring. In order to improve a system, it is important to know first where the organization stands initially. This will help to determine the quality improvement strategy (What objective can be chosen? What should be prioritized? How much effort will the change need?). The prevention of errors is also important. This shows a mindset turned toward a balanced system without various unexpected perturbations (like change in customer need, operator errors, capacity disequilibrium, etc.). A considerable category of goals is *all*. In this category, we grouped the tools that are mainly on global perspective in order to determine the way to proceed. Under that name are the methods of *kaizen*, DMAIC, DMADV, and DFSS, which are sequences of the work to do in order to reach successful improvements (Ishak et al., 2019). There are also the specialists (belt experts) and ERP systems, which are more resources than tools of primary importance for a QMS. ERP systems and specialists are a good symbol of integration of the quality preoccupations in the operational excellence management. This tool is part of the spine of companies in terms of management. Indeed, this goes further than just quality management. It encompasses also human resource management, for example. Also, the majority of the information flows in ERP. This represents well how quality management is part of a whole management system in order to reach operational excellence.

Moreover, concerning the type of improvement provided by the tools, global improvement tools are the most studied. The main tools in this category are DMAIC, 5S, *kaizen*, control charts, and Pareto Analysis. The tools do not require a high level of maturity in QMS. This is one of the reasons for the numerous studies done on it, as many cases exist and they have been documented for a long time. This also points that quality initiatives are desired but not understood enough to have important breakthrough on a very well-defined domain. Another important fact to notice is the small contribution of tools to prioritize the improvements. From the literature review, it has been found that the selection and prioritization of improvement projects is of primary importance in order to successfully implement a QMS. Therefore, a gap is identified between the knowledge of an important factor and the actual practice in the field. It underlines a second crucial factor, which is a trained population to act in QMS. This inconsistence between knowing a critical factor and not digging fully in its direction shows that people do not understand the topic well enough.

Finally, even if less documented, TOC and TQM are not useful for prioritization and statistics studies. As a matter of fact, TOC is a philosophy focused on removing existing bottlenecks, but does not reveal further achievable improvements. TQM is more a philosophy than a quantified tool. Its purpose is to gather the people around a quality objective more than to statistics on the activities going on in the organization.

Discussions on CSF, CFF, and Benefits

Many articles, like Patel and Desai (2018), Dedy et al. (2016), Lande et al. (2016), and Stankalla et al. (2018) list critical factors to be taken into account during the implementation of a QMS. These factors can be presented as success factors or as barriers to avoid during implementation. These lists are not exhaustive, but rather rank and present crucial ones. The same idea has been followed for this review study. The top 10 critical factors from the articles reviewed can be found in **Tables 2**, **4**. Often, the CSF are presented and the CFF are left on the side (Albliwi et al., 2014). Knowing how to successfully reach its goal also benefits from learning from the failures, as explained in Cannon and Edmondson (2005).

An important factor is the categorization of the articles mentioning each critical factor per philosophy focused on. Besides the rankings, some conclusions can be reached. SS is a philosophy based on statistics. It is the most quantified philosophy among LM, SS, L6S, TOC, and TQM. Therefore, the numbers are less subject to misinterpretation than policies and LM philosophies. This may explain why on the SS *Poor communication system* is weaker than on other philosophies. In addition, TOC seems mainly absent from the critical factors. This is mainly due to the fact that this philosophy has already proven some limits and is less studied then by the scientific community (Rahman, 1998; Ikeziri et al., 2019). Nevertheless, this philosophy still has a strong point in being defined well in terms of steps (Rahman, 1998; Kasemset, 2011; Ikeziri et al., 2019), which is a weakness presented in LM, SS, and L6S.

Moreover, regarding the benefits, they seem homogeneous on LM, SS and L6S. *Inventory reduction* is less cited for SS than for LM and L6S. This derives from the core of the definition of the philosophies. The LM aspect focuses on reducing the waste, while SS's main goal is to reduce the variability and not the quantity directly.

Finally, it is clearly visible that the critical factors and benefits mainly refer to the LM, SS and L6S more than the TOC and TQM. This is partly due to the trend present in the research. As explained previously in section Previous Literature, the research community lately have more focused on these three philosophies rather than on TOC and TQM. The method of ranking may be biased by these trends. An absence of article cited for a philosophy and factor may not mean that it is not important to be considered. A good opportunity to confirm the hypotheses relying on these rankings would be to survey the experts in quality management who are black belts and master black belts. They would be the most qualified people to address to.

Evolution Toward ZDM

The selection process for articles included in this literature review does not integrate ZDM. The idea of this section is more to link the findings from the review on LM, SS, L6S, TOC, and TQM with some important factors and new possibilities of ZDM. ZDM is a way of thinking of QMS with regards to product and process quality. It is based on a simple yet hard to achieve goal: Do right on the first attempts. For this reason, ZDM must be integrated into the production process right from the beginning, rather than trying to address the issues at a later stage and should follow a continuous improvement cycle based on standardized benchmarks. In fact, the standard SS methodology embraces ZDM as one of its core concepts, defining it as allowing a maximum of 3.4 defects per million products, since achieving zero defects in a real context is practically impossible. To achieve this, the evolution of Industry 4.0-enabling, data-driven innovation leads to an easier implementation of the ZDM concept, due to the availability of the required amount of data for techniques such as machine learning to work properly.

As explained in Psarommatis et al. (2019), the ZDM fulfills four missions: detect, repair, prevent, and predict. The first three missions are shared with the current quality philosophies.

Prediction, however, is new aspect. In fact, LM, SS, L6S, TOC, and TQM do not learn from defects. They just remove them. These philosophies analyze the past to improve in the future. Therefore, there is a loss of potentially important information from the present. Not analyzing the present creates an inertia between the occurrence of an event and the identification of an improvement linked to this event. One major change in ZDM is on the flow of information. Indeed, ZDM uses real-time data to prevent product from defect. Doing this, ZDM combines several quality control applications concerning production lines, machinery, automation applications, and supply chain processes. This is possible thanks to the development of IT systems and Industry 4.0. This helps to anticipate defects in order to fix them before too late. It is crucial to reach a state of early detection in order to have a sustainable system (Yusup et al., 2015). Moreover, this flow of information helps to better connect the global supply chain (Pagliosa et al., 2019), which is known to be a critical success factor for an efficient QMS. Another aspect on the predictive aspect is to predict defects not only in the product, but also in the process. In ZDM, the use of real-time data helps to dynamically monitor and tune the parameters in order to adapt the predictive maintenance. Downtime of a machine is known to be very costly. Reducing this downtime by a predictive maintenance of higher accuracy is a strong quality of ZDM (Dreyfus and Kyritsis, 2018).

In addition, Eleftheriadis and Myklebust (2016) have presented important aspects. First, a framework has been derived. This is an important point, since a critical failure factor for the current philosophies has been a lack of framework for implementation and systematic approach. This framework presents a systematic approach on the information data management. The idea is to dynamically deal with them in terms of real-time data to meet industry's new requirements so as to ensure a reliable, flexible, and sustainable system. Secondly, in this framework, corrections are autonomously dealt with. Therefore, the management team has less to focus on and can instead work on the human aspects. As pointed out in the critical factors, implementing a good culture of change is of primary importance and requires some time, newly provided by autonomous ZDM. Indirectly, this time combined with a fast information flow from customer online reviews helps to more quickly tackle the changes in customer needs and defaults of manufacturing that would have not been understood before.

Nevertheless, this connected flow of information exposes the organization to new risk and waste. Accordingly, this information must be secured (Seetharaman et al., 2019). The security department has to be trained for this new risk of data transformation and on how to prevent them. In addition, an accurate and fast data management system has to be established in order to avoid creating new waste. If not chosen with precaution, the monitored data may be very large. The processing time of this information increases with the volume of data. This could lead to some delay in predictive detection and the defect might have appeared. Moreover, a commonly used tool for QMS is JIT. As long as defects will be present in the quantities of items produced, a perfect JIT will be impossible, as some safety stock is necessary to compensate for these defects. Finally, an important weakness which has been pointed out is the lack of education of the organization's population. Implementing ZDM faces a similar barrier. A first step before inserting ZDM in QMS is to train actors in this QMS.

To conclude, hypotheses have been made and further research should be done in order to confirm or disprove them. Indeed, this discussion is only based on comparison aspects with the critical factors of LM, SS, L6S, TOC, TQM, and the trends concerning the new tools of information management.

CONCLUDING REMARKS AND FURTHER RESEARCH

In this multi-competitor market, an adequate QMS is essential to satisfy customer needs from a sustainable perspective. The implementation of this QMS relies on LM, SS, L6S, TOC, and TQM. These philosophies and mainly LM, SS, and L6S have proven to provide significant benefits like cost reduction, lead-time reduction, quality improvement, and more, when implemented in an appropriate manner. To achieve this implementation, some critical factors are to be taken into account like, a proper communication system, a visible top management commitment, a population trained in CI, and more, which have been described. An educated population is a key point to efficient improvement. Currently, the most-used tools that determine the current situation to prevent defects are flow mapping, global improvement, prioritization, quality improvement, statistical analysis, and time focus tools. Several goals can be achieved with many different tools. Nevertheless, due to a lack of understanding and a very large range of tools, the practitioner may be confused when choosing the one to use and miss other important ones. In that regard, an effort in education in universities and companies must be made. It can also be done thanks to quality institutions who lecture the experts. It is important to notice that institutions nowadays do not provide a standardized education on the quality management. Some work to standardize the knowledge to have in order to become a certified belt expert should be pursued. Also the research on tools should be pushed further in order to determine toolkits corresponding to several levels of maturity of quality management systems (Abdolshah and Jahan, 2006).

Finally, ZDM is an additionally more recent philosophy that is more and more enabled thanks to technological improvements. It allows a new goal, which is to predict defects. ZDM opens the gate for real-time data management to increase the efficiency of manufacturing organizations and to connect them better to the global supply chain. Nevertheless, the integration of these new technologies may raise new risks and waste. Indeed, the security of the information has to be ensured. The perimeter is then wider than physically securing the organization. This newly accessible flow of information can be so large that it may introduce a new sort of waste in the data management. Some research should be done to better identify and define them and how to reduce them.

- To conclude, this study investigated structured tools, critical factors, and benefits to give a better understanding of the topic. It has also provided insights on the new perspective offered by ZDM.
- From the findings of this article, some further steps should be followed in order to strengthen the understanding of QMS.
- Pursue research on tools used in order to propose a standardized toolkit corresponding to several level of maturity of the QMS.
- Standardize and develop the education on the improvement philosophies in order to increase the number of experts and ensure that they have the same level of knowledge.
- Confront the hypothetical findings on ZDM with experimental cases and experts point of view.

AUTHOR CONTRIBUTIONS

FP paper conceptualization and literature search, queries creation, and overall coordination of the project. FP and SP papers analysis and classification. FP, GM, and SP paper writing. DK paper review. All authors contributed to the article and approved the submitted version.

ACKNOWLEDGMENTS

The presented work presented supported by the projects Z-Bre4k and QU4LITY, EU H2020 projects under grant agreements Nos. 768869 and 825030 accordingly. The paper reflects the authors' views and the Commission is not responsible for any use that may be made of the information it contains.

REFERENCES

Abdolshah, M., and Jahan, A. (2006). "How to use continuous improvement tools in different life periods of organization," in *ICMIT 2006 Proceedings - 2006 IEEE International Conference on Management of Innovation and Technology*, Vol. 2 (Singapore), 772–777. doi: 10.1109/ICMIT.2006.262325

Albliwi, S., Antony, J., Abdul Halim Lim, S., and van der Wiele, T. (2014). Critical failure factors of lean six sigma: a systematic literature review. *Int. J. Qual. Reliab. Manage*. 31, 1012–1030. doi: 10.1108/IJQRM-09-2013-0147

Albliwi, S. A., Antony, J., and Lim, S. A. H. (2015). A systematic review of Lean six sigma for the manufacturing industry. *Bus. Process Manage. J*. 21, 665–691. doi: 10.1108/BPMJ-03-2014-0019

Alcaide-Muñoz, C., and Gutierrez-Gutierrez, L. J. (2017). Six Sigma and organisational ambidexterity: a systematic review and conceptual framework. *Int. J. Lean Six Sigma* 8, 436–456. doi: 10.1108/IJLSS-08-2016-0040

Alexander, P., Antony, J., and Rodgers, B. (2019). Lean Six Sigma for small- and medium-sized manufacturing enterprises: a systematic review. *Int. J. Qual. Reliab. Manage*. 36, 378–397. doi: 10.1108/IJQRM-03-2018-0074

Alhuraish, I., Robledo, C., and Kobi, A. (2015). "The effective of lean manufacturing and six sigma implementation," in *2015 International Conference on Industrial Engineering and Systems Management (IESM)* (Seville: IEEE), 453–460. doi: 10.1109/IESM.2015.7380197

Al-Khalili, A., and Subari, K. (2014). Understanding the importance of total quality management dimensions: Critical review of soft and hard aspects. *Int. J. Serv. Oper. Manage*. 18, 468–482. doi: 10.1504/IJSOM.2014.063246

Alsaffar, I., and Ketan, H. (2018). Reviewing the Effects of Integrated Lean Six Sigma Methodologies with Ergonomics Principles in an Industrial Workstation. *IOP Conf. Ser. Mater. Sci. Eng*. 433:012060. doi: 10.1088/1757-899X/433/1/012060

Alsmadi, M., and Khan, Z. (2010). "Lean sigma: the new wave of business excellence, literature review and a framework," in *2010 2nd International Conference on Engineering System Management and Applications, ICESMA 2010* (Sharjah).

Antony, J., Snee, R., and Hoerl, R. (2017). Lean six sigma: yesterday, today and tomorrow. *Int. J. Qual. Reliab. Manage*. 34, 1073–1093. doi: 10.1108/IJQRM-03-2016-0035

Arnheiter, E. D., and Maleyeff, J. (2005). The integration of lean management and six sigma. *TQM Mag*. 17, 5–18. doi: 10.1108/09544780510573020

Atmaca, E., and Girenes, S. S. (2011). Lean six sigma methodology and application. *Qual. Quant. 474*, 2107–2127. doi: 10.1007/s11135-011-9645-4

Bendell, T. (2006). A review and comparison of six sigma and the lean organisations. *TQM Mag*. 18, 255–262. doi: 10.1108/09544780610659989

Bhamu, J., and Singh Sangwan, K. (2014). Lean manufacturing: literature review and research issues. *Int. J. Oper. Prod. Manage*. 34, 876–940. doi: 10.1108/IJOPM-08-2012-0315

Cannon, M. D., and Edmondson, A. C. (2005). Failing to learn and learning to fail (intelligently): How great organizations put failure to work to innovate and improve. *Long Range Plann*. 38, 299–319. doi: 10.1016/j.lrp.2005.04.005

Chiarini, A. (2013). *Lean Organization: From the Tools of the Toyota Production System to Lean Office*. Springer Available online at: http://www.springer.com/business$+$%26$+$management/production/book/978-88-470-2509-7 (accessed February 17, 2020)

Costa, L. B. M., Godinho Filho, M., Fredendall, L. D., and Gómez Paredes, F. J. (2018). Lean, six sigma and lean six sigma in the food industry: a systematic literature review. *Trends Food Sci. Technol*. 82, 122–133. doi: 10.1016/j.tifs.2018.10.002

Dedy, A. N., Zakuan, N., Bahari, A. Z., Ariff, M. S. M., Chin, T. A., and Saman, M. Z. M. (2016). "Identifying critical success factors for tqm and employee performance in malaysian automotive industry: a literature review," in *IOP Conference Series: Materials Science and Engineering* (Bali: Institute of Physics Publishing). doi: 10.1088/1757-899X/131/1/012016

Dreyfus, P. A., and Kyritsis, D. (2018). "A framework based on predictive maintenance, zero-defect manufacturing and scheduling under uncertainty tools, to optimize production capacities of high-end quality products," in *IFIP Advances in Information and Communication Technology* (New York, NY: LLC Springer), 296–303. doi: 10.1007/978-3-319-99707-0_37

Edgeman, R. L., and Dugan, J. P. (2008). Six sigma from products to pollution to people. *Total Qual. Manag. Bus. Excell*. 19, 1–9. doi: 10.1080/14783360701601918

Eleftheriadis, R. J., and Myklebust, O. (2016). "A guideline of quality steps towards zero defect manufacturing in industry," in *2016 International Conference on Industrial Engineering and Operations Management* (Detroit, MI), 332–340.

Gamal Aboelmaged, M. (2010). Six Sigma quality: a structured review and implications for future research. *Int. J. Qual. Reliab. Manage*. 27, 268–317. doi: 10.1108/02656711011023294

Gillen, D. (2017). "Benchmarking and performance measurement: the role in quality management," in *Handbook of Logistics and Supply-Chain Management* (Emerald Group Publishing Limited), 325–338. doi: 10.1108/9780080435930-020

Gladysz, B., and Buczacki, A. (2018). Wireless technologies for lean manufacturing – a literature review. *Manag. Prod. Eng. Rev*. 9, 20–34. doi: 10.12783/dtetr/icpr2017/17575

Goldratt, M. E. (2020). *What Is This Thing Called Theory of Constraints and How Should It Be*. Available online at: https://books.google.ch/books/about/What_is_this_thing_called_theory_of_cons.html?id=FA8KAQAAMAAJ&redir_esc=y (accessed February 17, 2020)

Gupta, S., and Jain, S. K. (2013). A literature review of lean manufacturing. *Int. J. Manage. Sci. Eng. Manage*. 8, 241–249. doi: 10.1080/17509653.2013.825074

Hutchins, D. (2016). *Hoshin Kanri: The Strategic Approach to Continuous Improvement*, *1st Edn*. London: Routledge doi: 10.4324/9781315587035

Ikeziri, L. M., Souza, F. B., de, Gupta, M. C., and de Camargo Fiorini, P. (2019). Theory of constraints: review and bibliometric analysis. *Int. J. Prod. Res*. 57, 5068–5102. doi: 10.1080/00207543.2018.1518602

Ishak, A., Siregar, K., Asfriyati, and Naibaho, H. (2019). Quality control with six sigma DMAIC and grey failure mode effect analysis (FMEA): a review. *IOP Conf. Ser. Mater. Sci. Eng*. 505:012057. doi: 10.1088/1757-899X/505/1/012057

Ismail, M. Z. M., Zainal, A. H., Kasim, N. I., and Mukhtar, M. A. F. M. (2019). A mini review: lean management tools in assembly line at automotive industry. *IOP Conf. Ser. Mater. Sci. Eng*. 469:012086. doi: 10.1088/1757-899X/469/1/012086

Kasemset, C. (2011). "A review on quality improvement and Theory of Constraints (TOC)," in *2011 IEEE International Conference on Quality and Reliability* (Bangkok: IEEE), 327–330. doi: 10.1109/ICQR.2011.6 031735

Kedar, A. P., Lakhe, R. R., Deshpande, V. S., Washimkar, P. V., and Wakhare, M. V. (2008). "A comparative review of TQM, TPM and related organisational performance improvement programs," in *Proceedings - 1st International Conference on Emerging Trends in Engineering and Technology, ICETET* 2008, 725–730. doi: 10.1109/ICETET.2008.133

Kirche, E., and Srivastava, R. (2005). An ABC-based cost model with inventory and order level costs: A comparison with TOC. *Int. J. Prod. Res*. 43, 1685–1710. doi: 10.1080/002075412331317836

Kumar, M., and Antony, J. (2008). Comparing the quality management practices in UK SMEs. *Ind. Manage. Data Syst*. 108, 1153–1166. doi: 10.1108/02635570810914865

Kumar, P., Maiti, J., and Gunasekaran, A. (2018). Impact of quality management systems on firm performance. *Int. J. Qual. Reliab. Manage*. 35, 1034–1059. doi: 10.1108/IJQRM-02-2017-0030

Lande, M., Shrivastava, R. L., and Seth, D. (2016). Critical success factors for lean six sigma in smes (small and medium enterprises). *TQM J*. 28, 613–635. doi: 10.1108/TQM-12-2014-0107

Linderman, K., Schroeder, R. G., Zaheer, S., and Choo, A. S. (2003). Six Sigma: a goal-theoretic perspective. *J. Oper. Manage*. 21, 193–203. doi: 10.1016/S0272-6963(02)00087-6

Makwana, A. D., and Patange, G. S. (2019). A methodical literature review on application of Lean & six sigma in various industries. *Aust. J. Mech. Eng*. 1–5. doi: 10.1080/14484846.2019.1585225.

Martínez-Lorente, A. R., Dewhurst, F., and Dale, B. G. (1998). Total quality management: Origins and evolution of the term. *TQM Mag*. 10, 378–386. doi: 10.1108/09544789810231261

Martins, A. F., Costa Affonso, R., Tamayo, S., Lamouri, S., and Baldy Ngayo, C. (2015). "Relationships between national culture and lean management: a literature review," in *2015 International Conference on Industrial Engineering and Systems Management (IESM)* (Seville: IEEE), 352–361. doi: 10.1109/IESM.2015.7380183

Muraliraj, J., Zailani, S., Kuppusamy, S., and Santha, C. (2018). Annotated methodological review of Lean Six Sigma. *Int. J. Lean Six Sigma* 9, 2–49. doi: 10.1108/IJLSS-04-2017-0028

Nanda, V. (2005). *Quality Management System Handbook for Product Development Companies*. Boca Raton, FL: CRC Press. doi: 10.1201/97814200 25309

Nandurkar, K. N., Wakchaure, V. D., and Kallurkar, S. P. (2014). "The simulation-based comparison of joint implementation of JIT, TQM, TPM and SCM methods," in *Transactions of the North American Manufacturing Research Institution of SME* (Detroit, MI: Society of Manufacturing Engineers), 353–361.

Nithia, K., Noordin, M. Y., and Saman, M. Z. M. (2015). lean production weaknesses in manufacturing industry: a review. *Appl. Mech. Mater*. 735, 344–348. doi: 10.4028/www.scientific.net/AMM.73 5.344

Oke, S. A. (2007). *Six Sigma: A Literature Review*. Available online at: http://sajie.journals.ac.za (accessed February 17, 2020)

Pacheco, D., Pergher, I., Vaccaro, G. L. R., Jung, C. F., and ten Caten, C. (2015). 18 comparative aspects between Lean and Six Sigma: Complementarity and implications. *Int. J. Lean Six Sigma* 6, 161–175. doi: 10.1108/IJLSS-05-2014-0012

Pagliosa, M., Tortorella, G., and Ferreira, J. C. E. (2019). Industry 4.0 and Lean Manufacturing: A systematic literature review and future research directions. *J. Manuf. Technol. Manage*. doi: 10.1108/JMTM-12-2018-0446. [Epub ahead of print].

Patel, M., and Desai, D. A. (2018). Critical review and analysis of measuring the success of six sigma implementation in manufacturing sector. *Int. J. Qual. Reliab. Manage*. 35, 1519–1545. doi: 10.1108/IJQRM-04-2017-0081

Pinho, C., and Mendes, L. (2017). IT in lean-based manufacturing industries: systematic literature review and research issues. *Int. J. Prod. Res*. 55, 7524–7540. doi: 10.1080/00207543.2017.1384585

Psarommatis, F., May, G., Dreyfus, P.-A., and Kiritsis, D. (2019). Zero defect manufacturing: state-of-the-art review, shortcomings and future directions in research. *Int. J. Prod. Res*. 7543, 1–17. doi: 10.1080/00207543.2019.1605228

Rahman, S. (1998). Theory of constraints:a review of the philosophy and its applications. *Int. J. Oper. Prod. Manage*. 18, 336–355. doi: 10.1108/01443579810199720

Rothwell, W. J., Stavros, J. M., and Sullivan, R. L. (2015). *Practicing Organization Development : Leading Transformation and Change*. Hoboken, NJ: Wiley. doi: 10.1002/9781119176626

Ruben, R., Vinodh, B. S., and Asokan, P. (2018). Lean Six Sigma with environmental focus: review and framework. *Int. J. Adv. Manuf. Technol*. 94, 4023–4037. doi: 10.1007/s00170-017-1148-6

Seetharaman, A., Patwa, N., Saravanan, A. S., and Sharma, A. (2019). Customer expectation from Industrial Internet of Things (IIOT). *J. Manuf. Technol. Manage*. 30, 1161–1178. doi: 10.1108/JMTM-08-2018-0278

Singh, J., and Singh, H. (2012). Continuous improvement approach: atate-of-art review and future implications. *Int. J. Lean Six Sigma* 3, 88–111. doi: 10.1108/20401461211243694

Siregar, K., Ariani, F., Ginting, E., and Trie Dinda, M. P. (2019). Lean six sigma for manufacturing industry: a review. *IOP Conf. Ser. Mater. Sci. Eng*. 505:012056. doi: 10.1088/1757-899X/505/1/012056

Stankalla, R., Koval, O., and Chromjakova, F. (2018). A review of critical success factors for the successful implementation of lean six sigma and six sigma in manufacturing small and medium sized enterprises. *Qual. Eng*. 30, 453–468. doi: 10.1080/08982112.2018.1448933

Sundar, R., Balaji, A. N., and Kumar, R. M. S. (2014). A review on lean manufacturing implementation techniques. *Procedia Eng*. 97, 1875–1885. doi: 10.1016/j.proeng.2014.12.341

Talha, M. (2004). Total quality management (TQM): an overview. *Bottom Line* 17, 15–19. doi: 10.1108/08880450410519656

Tjahjono, B., Ball, P., Vitanov, V. I., Scorzafave, C., Nogueira, J., Calleja, J., et al. (2010). Six sigma: a literature review. *Int. J. Lean Six Sigma* 1, 216–233. doi: 10.1108/20401461011075017

Van Iwaarden, J., Van Der Wiele, T., Dale, B., Williams, R., and Bertsch, B. (2008). The Six Sigma improvement approach: A transnational comparison. *Int. J. Prod. Res*. 46, 6739–6758. doi: 10.1080/00207540802234050

Wilson, A., Zeithaml, V., Bitner, M. J., and Gremler, D. (2016). *Services Marketing: Integrating Customer Focus Across the Firm*. 3rd Europe.

Yamashina, H. (1995). Japanese manufacturing strategy and the role of total productive maintenance. *J. Qual. Maint. Eng*. 1, 27–38. doi: 10.1108/13552519510083129

Yusup, M. Z., Wan Mahmood, W. H., Salleh, M. R., Muhamad, M. R., and Saptari, A. (2015). Synthesizing the domain of lean practices in manufacturing operations: a review. *J. Teknol*. 77, 153–162. doi: 10.11113/jt.v77.4133

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

frontiers in Computer Science

ORIGINAL RESEARCH
published: 11 September 2020
doi: 10.3389/fcomp.2020.00041

Physically Inspired Data Compression and Management for Industrial Data Analytics

Ramin Sabbagh[1]*, *Zicheng Cai*[1], *Alec Stothert*[2] and *Dragan Djurdjanovic*[1]

[1] Walker Department of Mechanical Engineering, The University of Texas at Austin, Austin, TX, United States, [2] The MathWorks, Inc., Natick, MA, United States

With the huge and ever-growing volume of industrial data, an enormous challenge of how this data should be handled, stored, and analyzed emerges. In this paper, we describe a novel method that facilitates automated signal parsing into a set of exhaustive and mutually exclusive segments, which is coupled with extraction of physically interpretable signatures that characterize those segments. The resulting numerical signatures can be used to approximate a wide range of signals within some arbitrary accuracy, thus effectively turning the aforementioned signal parsing and signature extraction procedure into a signal compression process. This compression converts raw signals into physically plausible and interpretable features that can then be directly mined in order to extract useful information via anomaly detection and characterization, quality prediction, or process control. In addition, distance-based unsupervised clustering is utilized to organize the compressed data into a tree-structured database enabling rapid searches through the data and consequently facilitating efficient data mining. Application of the aforementioned methods to multiple large datasets of sensor readings collected from several advanced manufacturing plants showed the feasibility of physics-inspired compression of industrial data, as well as tremendous gains in terms of search speeds when compressed data were organized into a distance-based, tree-structured database.

Keywords: industrial data analytics, physically-interpretable data compression, industrial database organization, industrial database searching, industrial internet of things

OPEN ACCESS

Edited by:
Dimitris Kiritsis,
École Polytechnique Fédérale de Lausanne, Switzerland

Reviewed by:
Alexandros Bousdekis,
National Technical University of Athens, Greece
Hongbae Jun,
Hongik University, South Korea

***Correspondence:**
Ramin Sabbagh
sabbagh@utexas.edu

Specialty section:
This article was submitted to Mobile and Ubiquitous Computing, a section of the journal Frontiers in Computer Science

Received: *07 July 2020*
Accepted: *06 August 2020*
Published: *11 September 2020*

Citation:
Sabbagh R, Cai Z, Stothert A and Djurdjanovic D (2020) Physically Inspired Data Compression and Management for Industrial Data Analytics. Front. Comput. Sci. 2:41. doi: 10.3389/fcomp.2020.00041

INTRODUCTION

It is not widely known that industrial equipment already generates more data than computer and social networks, with almost double the growth rate, leading to tremendous amounts of pertinent data (Kalyanaraman, 2016). This provides an ever-growing opportunity to mine that data for useful information via e.g., prediction of outgoing product quality, process monitoring and control or optimization of operations.

Nevertheless, applications of Artificial Intelligence (AI) and Machine Learning (ML) in industry are lagging behind advancements in the realm of computer and social networks (Nasrabadi, 2007). The main reason is that the nature and characteristics of the data in physical processes or industrial internet of things (Gilchrist, 2016) are different from what we see in computer and social networks (Atzori et al., 2010). In the realm of computer science, information about events that are relevant to modeling and characterization of the underlying system are directly available in the data—e.g., who is talking to whom, for how long and what the relevant locations are, or which website you

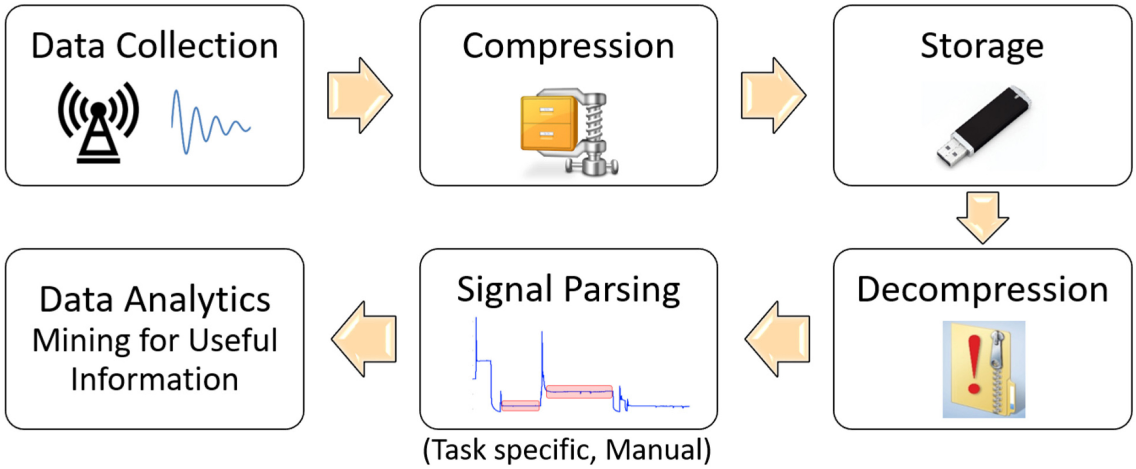


FIGURE 1 | Illustration of the traditional data processing and management for data mining in manufacturing.

are on, for how long and which website you are going to go after that, etc. On the other hand, in the industrial internet of things (IIoT), events are embedded in the data and are not directly visible. For instance, beginning and ending moments of a reaction in a chemical reactor, or moment and location of particle emission and trajectory of that particle in a semiconductor vacuum tool—all this information is not directly observable and is embedded in the signals emitted during the corresponding processes. Finding and characterizing such events in industrial data can link the mining of useful information from those signals to the realm of discrete mathematics and thus leverage tremendous advancements of AI and ML in the domains of computer and social networks. The work presented in this paper can be seen as an effort in the direction of establishing such a link.

At this moment, let us note that one of the main problems in utilizing the ever-growing volume of industrial data is the way that the data is handled at the very source. When it comes to sensor readings from manufacturing machines and equipment, industries tend to store the raw time-series (Kendall and Ord, 1990), with occasional use of various, usually lossless compression methods adopted from computer science in order to cope with the enormous data volumes (Sayood, 2002). These compression tools, such as run-length based compression (Hauck, 1986), Huffman compression (Tharini and Ranjan, 2009), delta compression appliance (Mogul et al., 2002), or the Lempel–Ziv–Welch (LZW) compression methods (Ping, 2002), are inherently designed to maximize compression rates, while minimizing information loss.

The purpose of the aforementioned compression tools is to turn raw signals into a set of coefficients that is much smaller than the original signal and is able to represent it perfectly, or very close to perfection, thus achieving compression and enabling storage of larger amounts of data. However, the resulting coefficients in the compressed domain do not have any relevance to the physical characteristics of the relevant processes and in order to perform mining of useful information from such data, one needs to decompress (reconstruct) the signal and extract the informative signatures out of it (Alves, 2018), as illustrated in **Figure 1**. Those informative signatures include metrics such as mean value, standard deviation, peak-to-peak values and other, usually statistics-inspired or expert-knowledge based quantities calculated for one or multiple signal portions deemed to be interesting for the data mining process[1].

Nevertheless, determination of the informative signal portions and relevant signatures involves a tremendous amount of expert process knowledge to insure the necessary information is indeed embedded in them (Djurdjanovic, 2018), which inherently makes this stage subjective and error prone. In addition, one is effectively blind to events in signal segments that were not selected for analysis, or to whatever is not depicted in the characteristics extracted from the raw signals. These drawbacks will be addressed in this paper by introducing a method for automated time-domain based segmentation of a signal into a set of exhaustive and mutually exclusive segments of steady state and transient behaviors, out of which we will extract a set of statistics-based and dynamics-inspired signatures that approximate the signal in those segments. Based on such signal segmentation and signatures extracted from each segment, one could approximately reconstruct the signal, which means that this procedure could be seen as a *signal compression tool*. In addition, physical interpretability of the newly proposed signal segmentation and signature extraction will enable mining for useful information about the underlying process directly in that compressed domain, without blind spots (segments) and without the need for human involvement in the process of signal parsing and extraction of signatures. **Figure 2** illustrates the novel data curation

[1]These signatures can be extracted from descriptions of relevant signal segments in various domains, such as time-domain, frequency-domain, or time-frequency-domain of signal representations (Chen and Lipps, 2000; Phinyomark et al., 2009; Suresh et al., 2013; Celler et al., 2019).

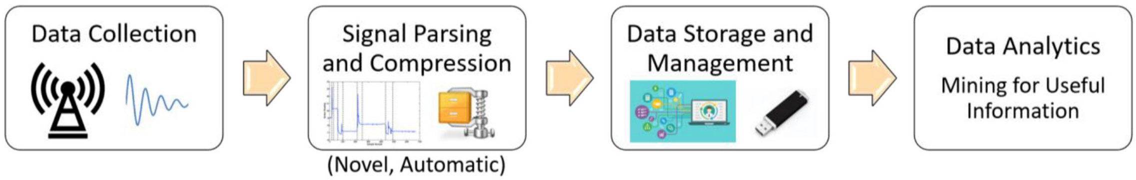


FIGURE 2 | Illustration of the newly enabled paradigm for data processing and management for data mining in manufacturing.

process that could be facilitated via the methods proposed in this paper.

More details can be found in the rest of the paper, which is organized as follows. In Methodology section, we will present the method for automated parsing of signals into a set of exhaustive and mutually exclusive steady state and transient segments, along with methods to characterize those segments using a set of physically interpretable signatures that facilitate approximate reconstruction of the signal and thus can be seen as its approximate compression. Furthermore, this section will present an approach to organize the compressed data into a distance-based tree structure that is much more efficient for search and retrieval than the temporally organized, list-based database structure traditionally utilized for industrial data. In Result section, we will present results of applying the newly introduced data compression and organization methods to sensor data gathered from several modern semiconductor manufacturing fabs. Finally, Conclusion and future work section gives conclusions of the research presented in this paper and outlines possible directions for future work.

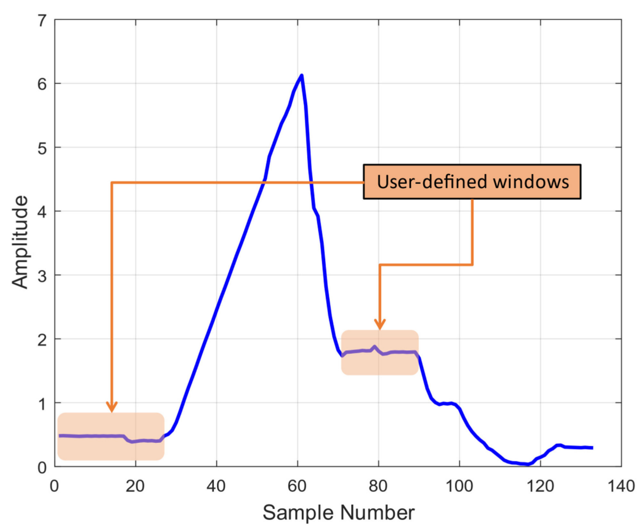


FIGURE 3 | Illustration of signal windowing utilized in traditional signal parsing and feature extraction.

METHODOLOGY

This section describes the novel physically-inspired data compression and management methodology. In Physics-inspired signal parsing & feature extraction for approximate signal compression section, the method for automated signal parsing and signature extraction will be explained, including a novel method for approximation-oriented physically-interpretable characterization of automatically detected transient portions of the signal. This signature extraction approach enables better reconstruction of the signal than what can be achieved using signal characterization based on only standard transient features described in IEEE 2011 (Pautlier et al., 2011). In Tree-structured data organization section, a distance-based tree-structured organization of industrial data will be proposed, enabling quick and accurate search of industrial databases directly in the compressed domain of coefficients extracted from the signals.

Physics-Inspired Signal Parsing & Feature Extraction for Approximate Signal Compression

Traditional signal parsing in the time domain is performed using human-defined windows based on physical knowledge of the process and human expertise. Such signal windows are selected usually because they correspond to a key portion of the process or are for whatever reason known or assumed to contain useful information. This often implies that the analysis ends up focusing on the steady state portions of the signals, where the processes actually take place. From these portions, a number of time and/or frequency domain signatures can be extracted, including mean, standard deviation, kurtosis, frequency peak locations and intensities, instantaneous frequency, group delay and so on. Consequently, large portions of the signal can be left unanalyzed, especially if the signal contains significant portions of transient behaviors (Kazemi, 1969; Hughes et al., 1979; Ramirez-Nunez, 2018; Yeap et al., 2018). **Figure 3** illustrates such traditional signal parsing based on user-defined windows, which leads to blind spots, redundancies in signatures and usually leaves out of the analysis process at least some (usually many or all) transient signal portions.

Recent publication (Haq et al., 2016) proposed a method for automatic segmentation of time-domain signal descriptions into a series of exhaustive and mutually exclusive segments of transient and steady state behaviors, as illustrated in **Figure 4**. From each steady state, statistics-inspired features, such as segment durations, or expected value and standard deviations of the sensor readings within the segment are extracted. On

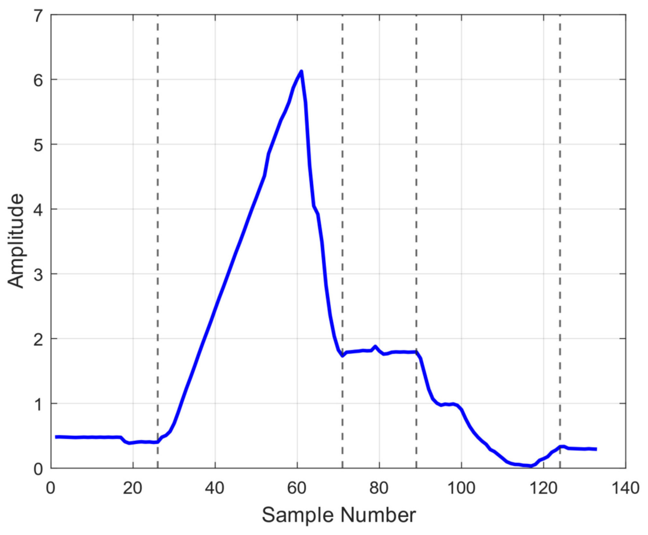


FIGURE 4 | Result of applying the automated signal parsing method introduced in (Haq et al., 2016) to the same signal as the one shown in **Figure 3**.

the other hand, from the transient portions, standard dynamics-inspired features, such as transition amplitudes, settling times, rise times, as well as post-shoot and pre-shoot features are derived (Pautlier et al., 2011). This ability for automated mining of the entire signal rather than only a selected subset of its portions led to great improvements in virtual metrology (Haq and Djurdjanovic, 2016) and defectivity analysis (Haq and Djurdjanovic, 2019) in advanced semiconductor manufacturing.

In addition, this automatic signal analysis opens the door to a significantly novel way of managing and utilizing densely sampled machine signals that are increasingly frequently encountered in modern industry. Namely, we can fully leverage the automatic parsing capabilities reported in (Haq et al., 2016) to enable encoding of a raw signal via a set of physically-interpretable statistical and dynamics-inspired signatures that compress the data into a domain which can be directly mined.

More specifically, within each transient segment, we can approximate the data using linear combination of sufficiently many complex exponential functions of the form

$$\hat{y}(t) = \sum\nolimits_{i=1}^{N} C_i \cdot e^{\lambda_i t} \tag{1}$$

where $\hat{y}(t)$ is the compressed transient model and for a given model order N, coefficients[2] C_i and λ_i can be determined using the well-known least squares fitting to the data. This form can be seen as a decomposition of a signal segment into contributions each of which can be associated with a dynamic mode of a linear differential equation (coefficients λ_i can be seen as roots of the characteristic polynomial of the differential equation that generated that segment, while coefficients C_i can be seen as strengths of the corresponding dynamic contribution to that segment). In this paper, the appropriate model order N in (1) was determined using Akaike Information Criterion[3] (AIC) (Sakamoto et al., 1986), though other information-theoretical or statistical approaches could be utilized for this purpose. Moreover, in addition to what was reported in (Haq et al., 2016), for each transient portion of the signal we also evaluated whether it can be better described as a single segment of form (1), or as a concatenation of two distinct segments of that form[4], with the more favorable option also selected using the AIC metric.

The original signals can be approximately reconstructed utilizing the signatures extracted from the steady state and transient segments. Specifically, in this paper, each steady state segment was approximated via the expected value of the amplitudes of the data points in that segment, while Equation (1) fit to any given transient section of the signal was used to approximate that signal portion.

In order to evaluate the efficacy of reconstructing the original signal from the compressed domain, adjusted R-squared (R^2) metric is utilized (Miles, 2014). Furthermore, in order to evaluate compression efficacy of our approach, we employ the intuitive metric expressing compression rate as

$$\text{Compression Rate} = 1 - \frac{\mathrm{N_C}}{\mathrm{L}} \tag{2}$$

where N_C is the number of coefficients (approximately) representing the signal in the compressed domain and L is the total length of the original signal.

In order to facilitate comparison and mining of the extracted signatures, one must ensure that corresponding signal segments populate consistent portions of the feature vector. Namely, though signals emitted by industrial processes usually have a fairly consistent structure, with a great majority of them having consistent number of segments, inherent process noise and inconsistencies could lead to situations when some of the signals have a slightly[5] larger or smaller number of segments, as compared to the majority of signals (Kosir and DeWall, 1994; Haq et al., 2016). For example, **Figure 5** shows signals emitted by the same sensor during processing of 4 distinct wafers in a semiconductor manufacturing tool. Signals A and B are two different signals with the segmentation form that appeared most frequently in that process, while Signal C contains two extra segments and Signal D has a missing segment. All these signals

[2] These coefficients can be real or complex.

[3] AIC is a well-known information-theoretical criterion that can be utilized to elegantly indicate when further increases in model complexity are not justified by the corresponding improvements in model accuracy.

[4] In a very similar manner, one could certainly explore possibilities to describe signal transients using concatenation of more than two segments of form (1). Nevertheless, our experience with real industrial data indicates that representing signal transients with up to two concatenated segments described by Eq. (1) led to excellent representation of a wide range of signals. This is why we only implemented a procedure that considers up to two distinct segments within each transient, though we once again acknowledge that a more general procedure should consider a more elaborate transient segmentation.

[5] For industrial processes, which are usually behaving in a fairly consistent manner, the difference in numbers of segments is small, with signals having at most a couple of extra or missing segments, and even such inconsistencies not appearing too often.

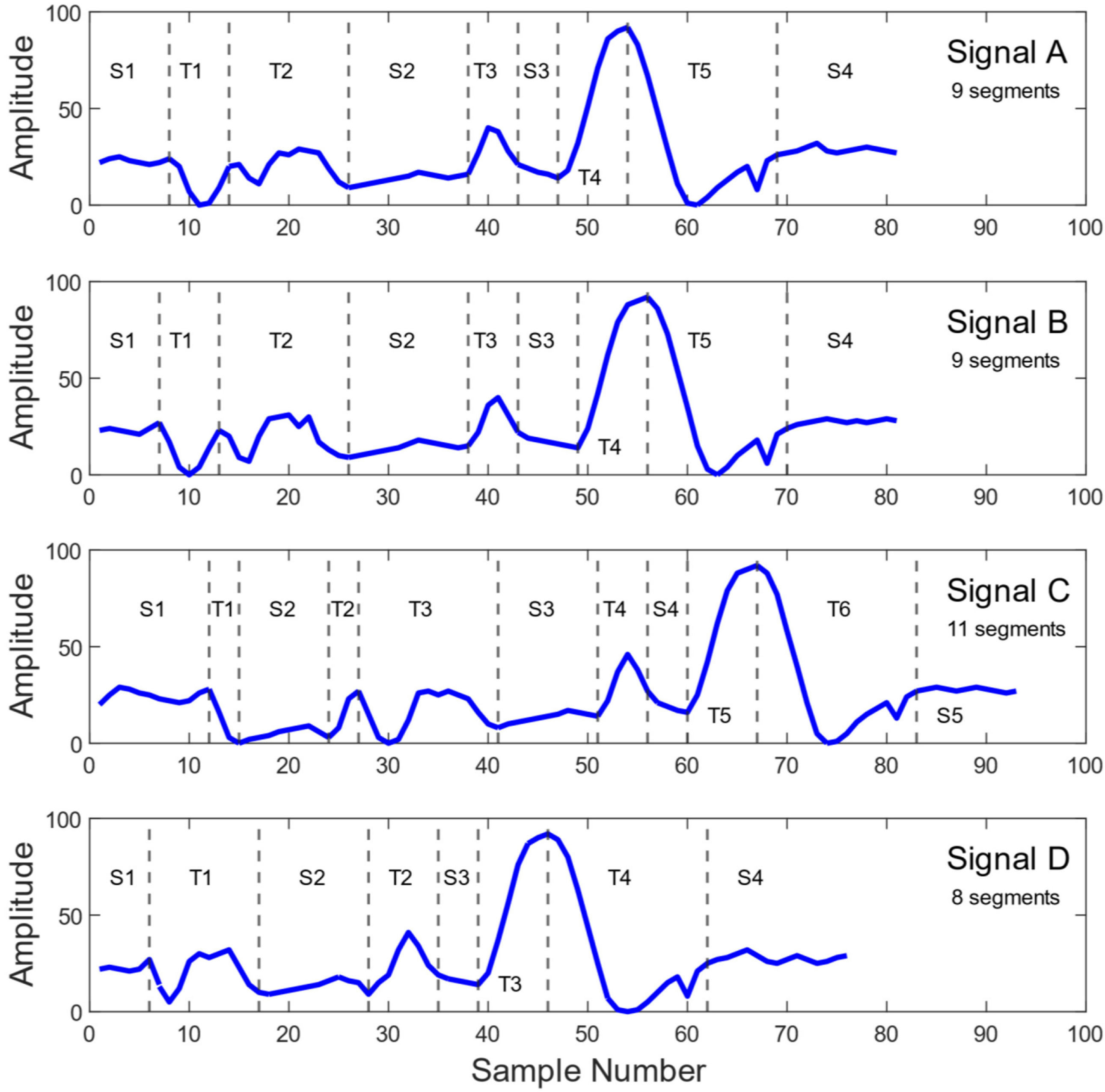


FIGURE 5 | Four distinct signals from a throttle valve angle sensor used in a Plasma Enhanced Chemical Vapor Deposition (PECVD) process operating in a major 300 mm wafer semiconductor fab. Signals **(A,B)** have the segmentation that appeared most frequently among the relevant process signals. Compared to signals **(A,B)**, signal **(C)** has one additional steady state segment and one additional transient segment. Signal **(D)** has one missing transient compared to signals **(A,B)**. Please note that steady state and transient segments are denoted as "S" and "T," respectively.

with different feature vectors need to be aligned in order to be consistently compared.

In order to perform the alignment of feature vectors, we identify all feature vectors in the available data that have the maximum number of segments and use them to create a *standard vector*, $\overrightarrow{SV}$, that has the same (maximal) number of steady-state and transient segments, each of which is characterized by averages of relevant coefficients, with any order inconsistencies within transient segments being resolved by adding appropriate number of zeros to lower order transients. For a feature vector that has fewer segments compared to the standard vector, we add appropriate placeholder segments to match the number and type of segments (steady state or transient) between that feature vector and the standard vector. Each steady state placeholder

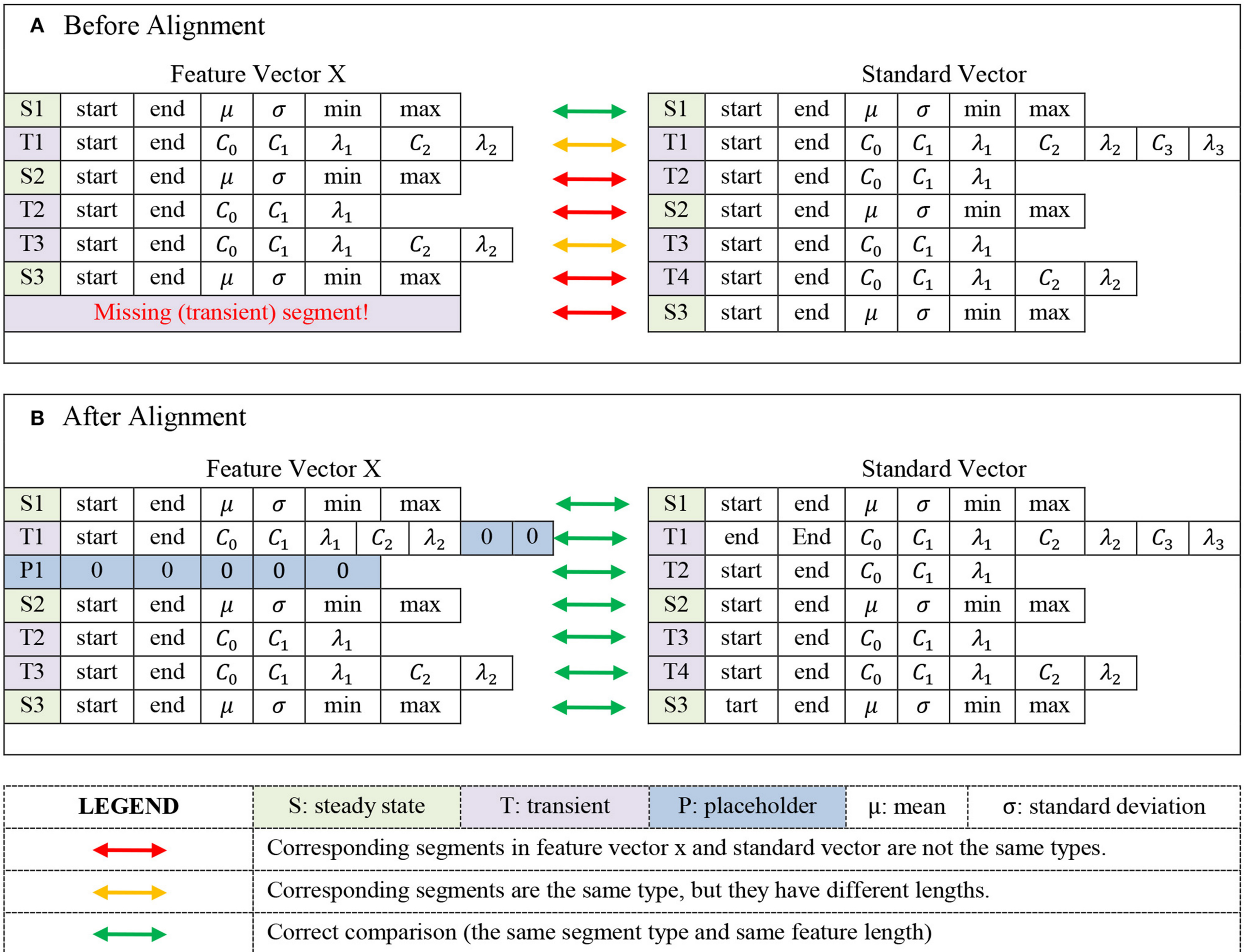


FIGURE 6 | Illustration of comparison between a feature vector with one missing segment and the standard vector before the alignment **(A)**, and after the alignment **(B)**. Based on segmentation of the standard vector, feature vector X misses a transient segment and hence, a placeholder corresponding to a transient segment should be inserted into it. Plot **(B)** illustrates how segments in the feature vector X are aligned with their corresponding segments in the standard vector by inserting a transient placeholder between segments T1 and S2 of the signal X, as well as by adding appropriate number of zeros to the set of C and λ coefficients in the T1 segment. Please note that insertion location for the transient placeholder is determined via the method described in sec Physics-inspired signal parsing & feature extraction for approximate signal compression.

segment was characterized by statistical characteristics set to zero (duration, mean, standard deviation, kurtosis, min, max, etc.), while each transient placeholder segment was characterized by the duration and all dynamic coefficients C_i and λ_i set to zero, with the number of dynamic coefficients in the transient placeholder depending on where in the signal it was inserted. The appropriate locations for placeholder insertions were found by evaluating all possibilities, with each candidate insertion option yielding a feature vector $\overrightarrow{FV}$ of the same dimensionality as the standard vector $\overrightarrow{SV}$, and its similarity to the standard vector being evaluated via the cosine between the two vectors

$$Alignment\ Similarity = \frac{\left\langle \overrightarrow{SV}, \overrightarrow{FV} \right\rangle}{\left\| \overrightarrow{SV} \right\| \left\| \overrightarrow{FV} \right\|} \tag{3}$$

where $\langle \cdot, \cdot \rangle$ denotes Euclidean inner product between two vectors and $\|\cdot\|$ denotes the corresponding vector norm. The option that yielded the highest similarity metric (3), i.e., insertion option that ended being the most co-linear with the standard vector was ultimately chosen. **Figure 6** demonstrates how the proposed alignment methodology is performed in a situation where a feature vector with a missing transient segment is being aligned to the standard vector[6].

It should be noted that as new data arrives, one could observe new feature vectors containing extra segments compared to the standard vector. In that case, the standard vector can be updated and all previously collected and aligned feature vectors would

[6]Please note that in **Figure 6**, FV and SV associated with the Eq. (3) are the concatenated vectors containing all the extracted feature from feature vector X and the standard vector respectively.

then need to be realigned based on the new standard vector. This could obviously be a rather computationally involved process, especially in large datasets. Nevertheless, our experience with real industrial data indicates that after a certain amount of data, the standard vector settles and rarely gets changed. Hence, it is recommended to perform feature vector alignments using sizeable initial datasets in order to make arrivals of feature vectors with extra segments less likely.

Tree-Structured Data Organization

Indexed databases with tree-based structures can be searched with logarithmic gains over databases organized as lists (Ramakrishnan and Gehrke, 2000). This is well-known within the computer science community, but is less known within the general engineering, and especially manufacturing research and practice communities. Recently, Aremu et al. (2018) suggested what is essentially a tree-based organization of industrial databases for the purpose of data curation for condition monitoring. The authors propose a hierarchical organization of the industrial data based on a number of criteria, including the underlying equipment condition and behavior modes. Nevertheless, the details of how to differentiate those condition and behavior modes when such information is not explicitly visible in the data, which is usually the case in real-life industrial processes, was not discussed. In addition, the authors did not discuss nor demonstrate quantitative benefits of such data organization.

To that end, in this paper, we propose the use of unsupervised clustering to autonomously identify underlying operating modes and conditions that are embedded in the physically-interpretable signatures obtained via compression of equipment sensor readings described in the previous section. Specifically, we use a Fritzke's growing gas based Growing Self-Organizing Map (GSOM) (Fritzke, 1994, 1995) to represent a given database of equipment sensor signatures via an appropriate number of clusters of data entries that are near to each other, as expressed via some distance matric[7]. GSOM-based clustering is accomplished through growth and adaptation of so-called weight vectors that tessellate the underlying data space into Voronoi sets, each of which consists of points that are nearest to a specific weight vector in the GSOM (Kohonen, 1990). Each cluster is formed by data entries that are inside a specific Voronoi set, which means that the data inside a cluster are closer to the weight vector associated with that cluster than any other weight vector in the GSOM. Following abundant research in machine condition-monitoring (Siegel and Lee, 2011; Lapira et al., 2012; Siegel, 2013; Hendrickx et al., 2020), clusters yielded by unsupervised clustering of equipment sensor signatures, such as those extracted through physically-interpretable compression described in section Physics-inspired signal parsing & feature extraction for approximate signal compression, can be seen as representative of the underlying equipment condition and operating regimes, and can thus serve as the foundation for the hierarchical tree-based organization of databases of those signatures.

Figure 7 illustrates the structure of such a database. Searching within it would consist of first identifying the nearest GSOM weight vector, thus identifying the cluster of entries similar to the query entry, after which only entries inside that cluster should be searched rather than the entire database. Of course, with large size databases, the number of clusters in the GSOM could grow as well, leading to the possibility to cluster the weight vectors (clusters) themselves and facilitate a multi-level tree-based database, as reported in (Sabbagh et al., 2020). Generally, such a "divide and conquer" approach that focuses the search onto areas of the database that are similar to the query item rather than exploring the whole database is the key factor enabling logarithmic acceleration of searches within such hierarchical, tree-based databases (Chow and Rahman, 2009).

The abovementioned acceleration, however, does come with some costs. Namely, if a query items falls close to the boundary of a Voronoi set (i.e., close to the boundary of a cluster), then some database entries similar to it could reside in the neighboring cluster or clusters. Search that focuses only on the cluster to which that query item belongs (i.e., only to the cluster corresponding to the weight vector nearest to the query item) will miss entries that reside in the neighboring clusters, which leads to deteriorated search precision and recall metrics (Buckland and Gey, 1994; Bhattacharya, 2014). These problems are well-known in computer science, which is why searches in tree-based databases can be augmented by expanding the search to database sections in the neighborhood of the section identified in the initial stages of the search (leaves of the database tree that are in the neighborhood of the tree leaf to which the search initially focuses). Consequently, in this paper, we explored possibilities to search database clusters in the topological neighborhood of the cluster identified by the nearest (best matching) GSOM weight vector (Balaban, 1982). Such expanded search takes longer time to accomplish, but it improves the search precision and recall metrics.

RESULTS

The newly proposed data compression methodology described in Physics-inspired signal parsing & feature extraction for approximate signal compression section of this paper was evaluated on two large datasets. One of those datasets (Dataset A) includes sensor readings obtained from a Plasma Enhanced Chemical Vapor Deposition (PECVD) process performed on a 300 mm wafer tool operating in a major semiconductor manufacturing fab. Sensor readings in Dataset A were collected from 50 different sensors at a 10 Hz sampling rate during production of over 45,000 wafers. The other dataset (Dataset B) contains sensor readings emitted by a 300 mm wafer plasma etch tool operating in another high-volume semiconductor manufacturing fab. Dataset B contains readings from 110 different sensors collected at 5 Hz during etching of 4,500 wafers.

[7]E.g. Euclidean, Mahalanobis, Manhattan or some other distance metric. Furthermore, please note that one same database can be indexed in multiple ways, using GSOM-based clustering based on different distance metrics. This would yield multiple sets of centroids (database keys) that parse that database and facilitate acceleration of searches.

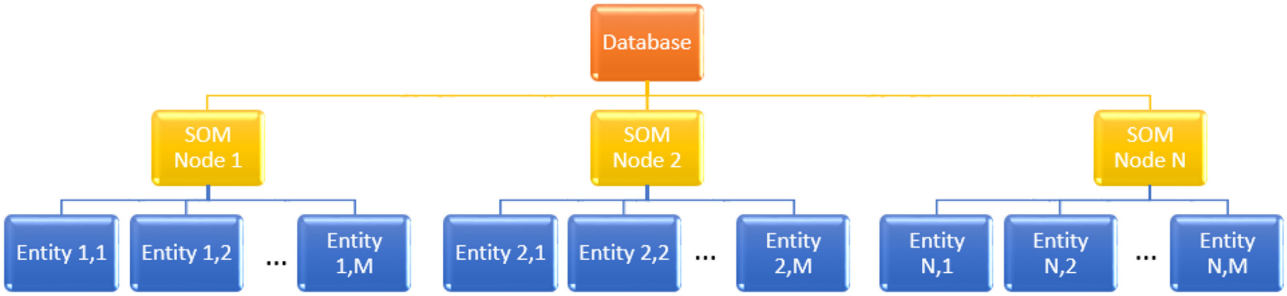


FIGURE 7 | Tree-based database organization enabled by unsupervised clustering of equipment signatures implemented using self-organizing maps (SOM).

TABLE 1 | Performance metrics associated with signal reconstruction.

Experimental Results	Performance Measures	Dataset A	Dataset B
All signals	Average Adjusted R^2	0.926	0.987
	Minimum Adjusted R^2	0.737	0.879
	Maximum Adjusted R^2	0.994	0.998
	Average Processing Time (s)	5.21 (s)	119 (s)
	Average Compression Rate (%)	53.94 %	71.26%

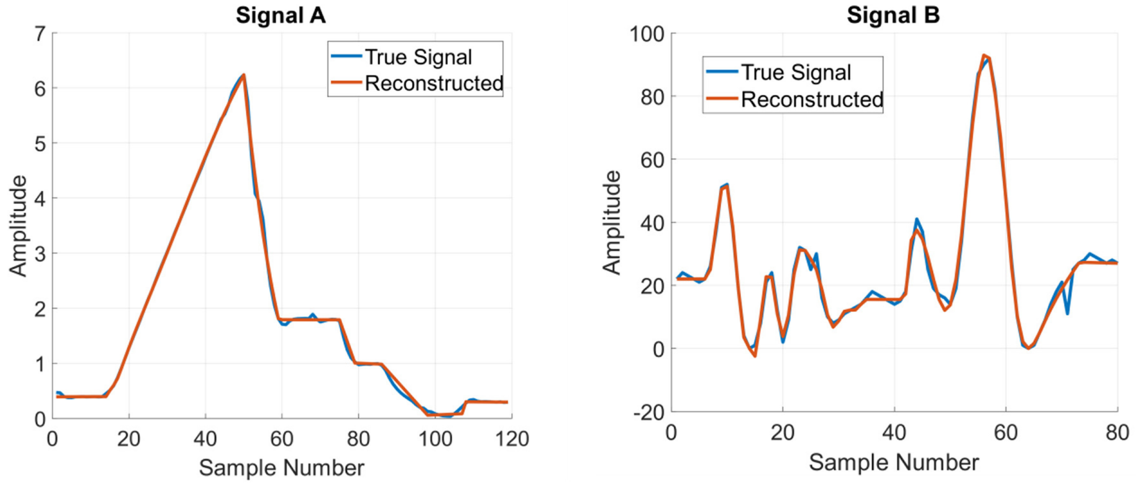


FIGURE 8 | Examples of the original and reconstructed signals for one pressure sensor reading (Signal **A**) and one throttle valve angle sensor reading (Signal **B**) observed in the Dataset A. The compression rate for the pressure sensor reading on the left of this figure was 0.6975, while the R-Squared metric for the reconstructed signal was 0.9987. Compression rate for the throttle valve signal on the right of this figure was 0.3355, while the R-Squared for the reconstructed signal was 0.9846.

Table 1 summarizes key metrics characterizing the compression rates and signal reconstruction performance in the relevant datasets. In terms of computational times[8], average time to process all signals relevant to a single wafer was 5.2 s in Dataset A, and 119 s in Dataset B. For illustration purposes, **Figure 8** shows two examples of original and reconstructed signals, with corresponding signal compression and reconstruction metrics. Both signals in **Figure 8** were taken from Dataset A.

Furthermore, Dataset A was large enough to realistically evaluate benefits of the distance-based data organization methodology proposed in Tree-structured data organization section. Signatures extracted via compression of the initial 1,000 signals from the Dataset A were clustered using Fritzke's growing gas GSOM method to yield the initial tree-based data organization. From the remaining 46,000 wafers, we randomly selected 200 wafers and for each vector of signatures extracted

[8]The times reported here correspond to processing on a regular Personal Computer with 32.0 GB RAM and a 6-core Intel® Xeon® CPU E5-1650 v4 @ 3.60GHz processor.

from signals emitted during manufacturing of those wafers, we evaluated the search performance of finding 10 nearest neighbors in the database. Such queries of industrial databases are of paramount importance for e.g., condition monitoring, where one needs to rapidly and correctly identify sensory signatures in the database that look most alike the newly observed (query) signature.

Growth and updating of the database were simulated by adding compressed sensor signatures from successive batches of 1,000 wafers from the Dataset A and adapting the GSOM to facilitate clustering and subsequent tree-based organization of the ever-growing dataset. **Figure 9** shows evolution of the number of clusters of the resulting SOM, while **Figure 10** shows computational times it took the GSOM to settle after

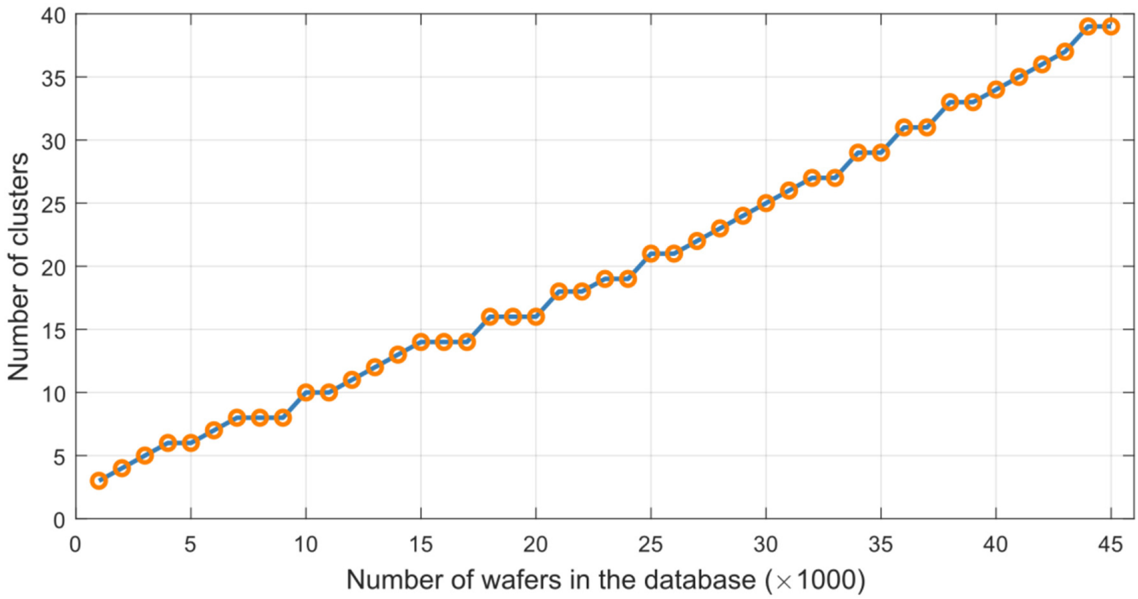


FIGURE 9 | Number of GSOM clusters in unsupervised clustering-based organization of the database of compression signatures obtained from the Dataset A.

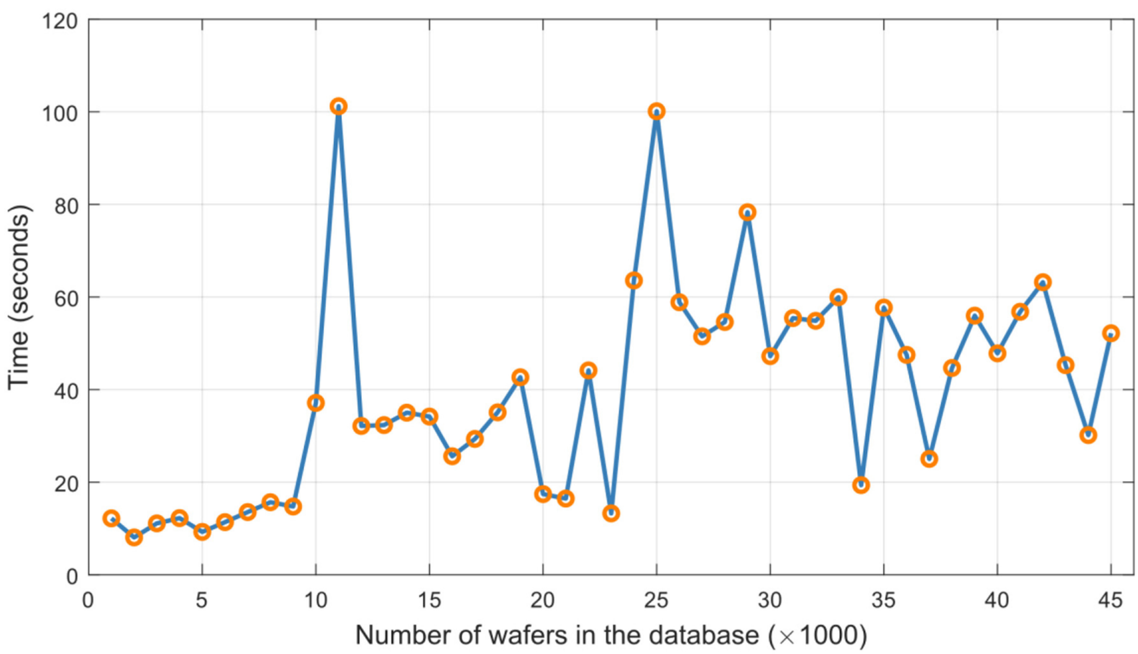


FIGURE 10 | Behavior of elapsed times needed for adjustments of the database of signatures obtained using newly proposed physically-interpretable compression of signals from Dataset A.

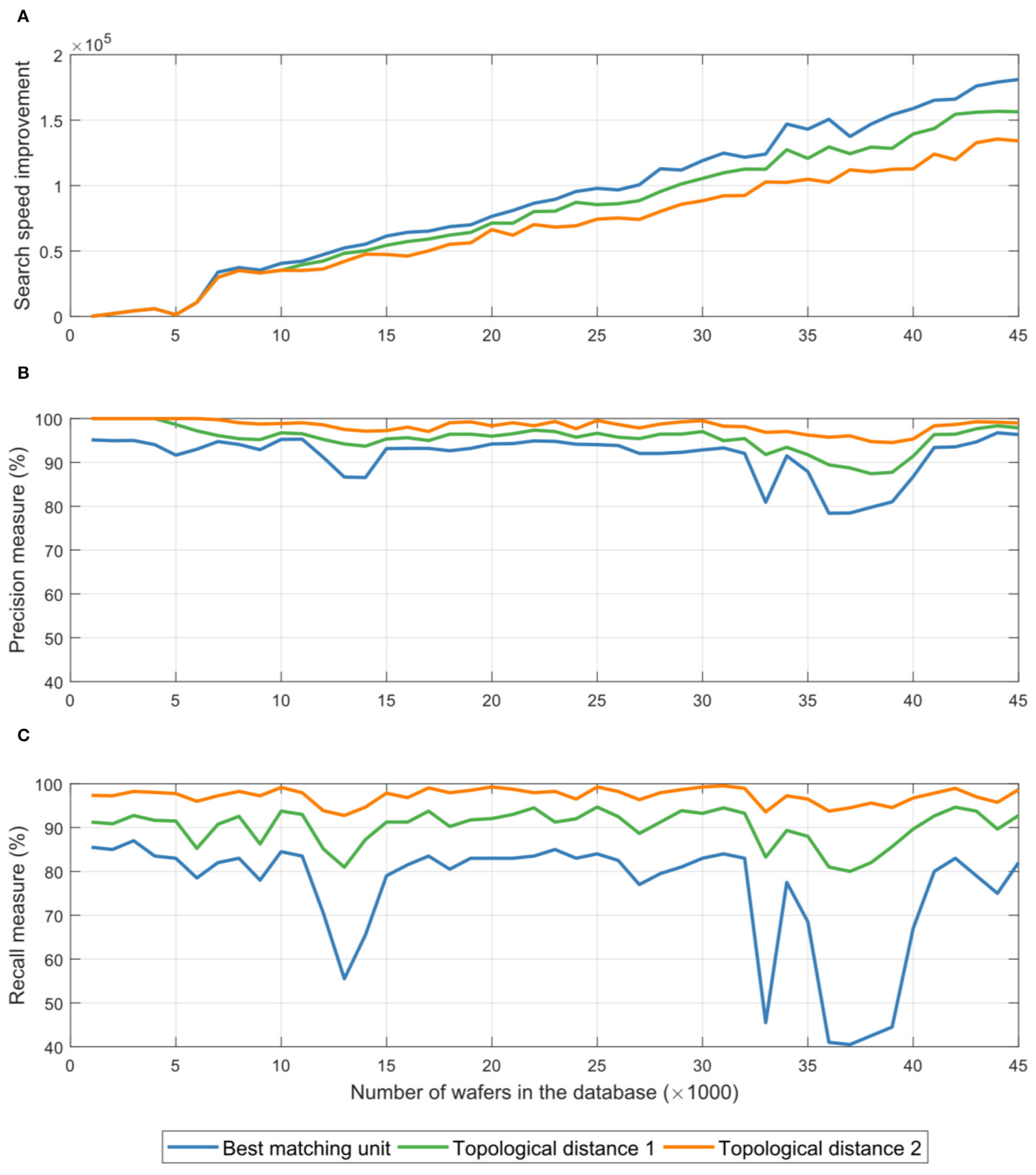


FIGURE 11 | Plot **(A)** shows the average ratio of times needed to search the GSOM-based database and times needed to conduct the corresponding search in the traditional, temporally organized database on the form of a list. Plots **(B,C)**, respectively, show the average precision and recall metrics of those searches. All tests were evaluated for searches that involved only the cluster corresponding to the weight vector nearest to the query item (blue curves), clusters in the immediate neighborhood of the one corresponding to the weight vector nearest to the query item (green curves), and clusters whose topological distance away from the one corresponding the weight vector nearest to the query item is less or equal to two (orange curves).

each introduction of signals from 1000-wafer batches[9]. Each time compressed signal signatures from a new batch of 1000 wafers were added to the database and the GSOM adaptation stopped, we again randomly selected 200 wafers that were not yet presented to the database and for each vector of compressed sensory signatures obtained from those wafers, we evaluated the search performance in finding 10 nearest neighbors in the database.

Search performance was evaluated using average precision and recall metrics, as well as the average speed of those searches for the cases when only the nearest database cluster was searched, as well as when GSOM clusters with topological distance 1 and 2 from the nearest cluster were searched[10]. **Figure 11A** shows the average ratio of times needed to search the GSOM-based database and times needed to conduct the corresponding search in the traditional, temporally organized database on the form of a list, while **Figures 11B,C,** respectively, show the average precision and recall metrics of those searches.

As expected, one can see from **Figure 11** that expanding the search into neighboring regions of the tree-based database leads to improved precision and recall metrics (more accurate search results). It can also be seen that these improvements come at the cost of additional times needed to conduct the searches. Nevertheless, it is clear that the tree-based database organization consistently yields search-time improvements that grow with the size of the database, with expanded searching of the tree slightly slowing the search, while delivering nearly perfect search results (average precision above 98% and average recall above 97% when one searches GSOM clusters, i.e., segments of the database tree, with topological distance 2 away from the best matching cluster).

Conclusion and Future Work

This paper presents an automated method for approximate compression of a signal based on extracting physically interpretable signatures from its time-domain description. Thus, the proposed data compression approach is appropriate for signals for which useful information is embedded directly in the time domain. Those are usually sensors of thermofluidic variables, such as flow, temperature, and pressure sensor readings in semiconductor tools, petrochemical plants, or pharmaceutical industries. However, in signals for which information is embedded in the frequency or joint time-frequency domain, such as vibrations from gearboxes and bearings, or acoustic emissions signals from cutting tools in machining, or civil engineering structures, this compression method is not appropriate.

The newly proposed method converts raw signals into signatures that can then be directly used for mining of useful information via e.g., detection and characterization of anomalies, quality prediction, or process control. The method also reduces the data storage burden since the signal could be approximately reconstructed from the extracted signatures. In addition, an unsupervised clustering method was used to organize the compressed data into a distance-based, tree-structured database. Such tree-based data organization is known in computer science to offer significant advantages in terms of speeding up searches in large databases. The benefits of the proposed methodology for data compression and organization were evaluated utilizing two large datasets from modern semiconductor manufacturing fabs. The results illustrate the feasibility of the aforementioned data compression method, as well as superior performance in terms of speed and accuracy of data searches in the newly proposed database structure, compared to searches in the conventionally organized industrial databases in the form of temporal lists.

Methodologically, a natural next step forward in this research would be to explore the possibilities of enabling physically plausible signal compression methodology in the frequency and time-frequency-domains. Such capabilities could be of tremendous benefits for condition-monitoring applications in rotating machinery and other mechanical systems. Another direction for future work should be the use of more powerful and general distance-based measures to organize the compressed database. e.g., stochastic automata (Eilenberg, 1974) could be used to yield alternative distance measure to determine the "similarity" between data points. which would greatly improve one's ability to compare signals even when novel segments that were not previously seen appear in the signal, or a segment that is usually there, but does not appear in the newly observed signal.

From the practical point of view, the methods described in this paper can be developed and implemented in an actual industrial setting, where the sheer volume of data represents a unique challenge. e.g., a modern semiconductor fabrication facility processing 300 mm wafers can stream well over 100 K signals, each of which can be (is) sampled at 10 Hz or higher. Effectively enabling data curation capabilities described in this paper in such a setting requires methodologies and solutions that intricately and innovatively link the software implementing data curation methodologies with the hardware that enables moving and processing of such enormous amounts of data. Such solutions require highly interdisciplinary skills in both hardware and software and their pursuit is outside the scope of this paper.

DATA AVAILABILITY STATEMENT

The data analyzed in this study is subject to the following licenses/restrictions: The datasets were given to our research group by semiconductor manufacturing companies for the research purposes only. Requests to access these datasets should be directed to dragand@me.utexas.edu.

[9]These times correspond to the so-called computational overhead needed to maintain the tree-based database organization (Bhattacharya, 2014). In the case of a list-based database organization, this time is essentially zero, since there are no adaptations needed to maintain the database organization.

[10]Such searching of database segments (bins) in the neighborhood of the initial search focus within a tree-based database is yet another common practice utilized to improve accuracy of database search results (Bhattacharya, 2014).

AUTHOR CONTRIBUTIONS

RS was the leading author coded majority of the methodologies, produced results, and participated in the writing of the manuscript. ZC was the first graduate student involved in this research, who implemented the AIC based identification of orders of transient segments, and conducted first search speed tests with SOM based database organization. AS contributed with research advising of graduate students and participated in the drafting and proofreading of the manuscript. DD defined the research topic, led the underlying research and participated in the drafting, and proofreading of the manuscript. All authors contributed to the article and approved the submitted version.

FUNDING

This work was supported in part by the National Science Foundation (NSF) under Cooperative Agreement No. EEC-1160494 and Cooperative Agreement No. IIP-1266279. This work is also supported in part by a membership of The MathWorks, Inc. in the University of Texas at Austin campus of the NSF Industry-University Cooperative Research Center on Intelligent Maintenance Systems. Any opinions, findings and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the NSF, or The MathWorks, Inc.

REFERENCES

Alves, V. (2018). *System and Methods for in-Storage on-Demand Data Decompression*. Google Patents.

Aremu, O. O., Salvador Palau, A., Hyland-Wood, D., Parlikad, A. K., and McAree, P. R. (2018). Structuring data for intelligent predictive maintenance in asset management. *IFAC-PapersOnLine*. 51, 514–519. doi: 10.1016/j.ifacol.2018.08.370

Atzori, L., Iera, A., and Morabito, G. (2010). The internet of things: a survey. *Comput. Netw*. 54, 2787–2805. doi: 10.1016/j.comnet.2010.05.010

Balaban, A. T. (1982). Highly discriminating distance-based topological index. *Chem. Phys. Lett*. 89, 399–404. doi: 10.1016/0009-2614(82)80009-2

Bhattacharya, A. (2014). *Fundamentals of Database Indexing and Searching*. New York, NY: CRC Press. doi: 10.1201/b17767

Buckland, M., and Gey, F. (1994). The relationship between recall and precision. *J. Am. Soc. Inf. Sci.* 45, 12–19. doi: 10.1002/(SICI)1097-4571(199401)45:1<12::AID-ASI2>3.0.CO;2-L

Celler, B. G., Le, P. N., Argha, A., and Ambikairajah, E. (2019). GMM-HMM based blood pressure estimation using time domain features. *IEEE Trans. Instrum. Meas*. 69, 3631–3641. doi: 10.1109/TIM.2019.2937074

Chen, V. C., and Lipps, R. D. (2000). Time frequency signatures of micro-Doppler phenomenon for feature extraction. *Wavelet Appl.* 4056, 220–226. doi: 10.1117/12.381683

Chow, T. W. S., and Rahman, M. K. M. (2009). Multilayer SOM with tree-structured data for efficient document retrieval and plagiarism detection. *IEEE Trans. Neural Netw*. 20, 1385–1402. doi: 10.1109/TNN.2009.2023394

Djurdjanovic, D. (2018). "Condition Monitoring and Operational Decision-Making in Modern Semiconductor Manufacturing Systems," in *Proceedings of the Pacific Rim Statistical Conference for Production Engineering* (Seoul), 41–66. doi: 10.1007/978-981-10-8168-2_5

Eilenberg, S. (1974). *Automata, Languages, and Machines*. New York, NY; London: Academic press.

Fritzke, B. (1994). Growing cell structures—a self-organizing network for unsupervised and supervised learning. *Neural Netw.* 7, 1441–1460. doi: 10.1016/0893-6080(94)90091-4

Fritzke, B. (1995). A growing neural gas network learns topologies. *Adv. Neural Inform. Proc. Syst.* 7, 625–632.

Gilchrist, A. (2016). *Industry 4.0: the Industrial Internet of Things*. A press. doi: 10.1007/978-1-4842-2047-4_10

Haq, A. A. U., Wang, K., and Djurdjanovic, D. (2016). Feature construction for dense inline data in semiconductor manufacturing processes. *IFAC-PapersOnLine* 49, 274–279. doi: 10.1016/j.ifacol.2016.11.047

Haq, A. U., and Djurdjanovic, D. (2016). Virtual metrology concept for predicting defect levels in semiconductor manufacturing. *Procedia CIRP* 57, 580–584. doi: 10.1016/j.procir.2016.11.100

Haq, A. U., and Djurdjanovic, D. (2019). Dynamics-inspired feature extraction in semiconductor manufacturing processes. *J. Ind. Inf. Integr.* 13, 22–31. doi: 10.1016/j.jii.2018.12.001

Hauck, E. L. (1986). *Data Compression Using Run Length Encoding and Statistical Encoding*. Google Patents.

Hendrickx, K., Meert, W., Mollet, Y., Gyselnck, J., Cornelis, B., Gryllias, K., et al. (2020). A general anomaly detection framework for fleet-based condition monitoring of machines. *Mech. Syst. Signal Process.* 139:106585. doi: 10.1016/j.ymssp.2019.106585

Hughes, T. J. R., Pister, K. S., and Taylor, R. L. (1979). Implicit-explicit finite elements in nonlinear transient analysis. *Comput. Methods Appl. Mech. Eng.* 17, 159–182. doi: 10.1016/0045-7825(79)90086-0

Kalyanaraman, S. (2016). *Industry 4.0 meets Cognitive IoT: Internet of Things Blog*. Available online at: https://www.ibm.com/blogs/internet-of-things/industry-4-0-meets-cognitive-iot/

Kazemi, H. (1969). Pressure transient analysis of naturally fractured reservoirs with uniform fracture distribution. *Soc. Pet. Eng. J.* 9, 451–462. doi: 10.2118/2156-A

Kendall, M. G., and Ord, J. K. (1990). *Time-Series 296*. Edward Arnold London.

Kohonen, T. (1990). Self-organizing map. *Proc. IEEE* 78, 1464–1480. doi: 10.1109/5.58325

Kosir, P., and DeWall, R. (1994). "Feature alignment techniques for pattern recognition," in *Proceedings of National Aerospace and Electronics Conference (NAECON'94)* (Dayton, OH), 128–132.

Lapira, E., Brisset, D., Ardakani, H. D., Siegel, D., and Lee, J. (2012). Wind turbine performance assessment using multi-regime modeling approach. *Renew. Energy* 45, 86–95. doi: 10.1016/j.renene.2012.02.018

Miles, J. (2014). R squared, adjusted R squared. *Wiley StatsRef Stat. Ref. Online*. doi: 10.1002/9781118445112.stat06627

Mogul, J., Krishnamurthy, B., Douglis, F., Feldmann, A., Goland, Y., van Hoff, A., et al. (2002). *Delta Encoding in HTTP*. Gennaio: IETF. 65, doi: 10.17487/rfc3229

Nasrabadi, N. M. (2007). Pattern recognition and machine learning. *J. Electron. Imaging* 16:49901. doi: 10.1117/1.2819119

Pautlier, N. G., Antonellis, J., Balestrieri, E., Blair, J., Calvin, J., Dallet, D., et al. (2011). *IEEE Std. 181-2011-IEEE Standard for Transitions, Pulses, and Related Waveforms (Revision of IEEE Std. 181-2003)*.

Phinyomark, A., Limsakul, C., and Phukpattaranont, P. (2009). A novel feature extraction for robust EMG pattern recognition. *arXiv Prepr. arXiv* 1, 71–80. Available online at: https://arxiv.org/abs/0912.3973v2

Ping, W. (2002). Realization and research of LZW lossless compression algorithm. *Comput. Eng.* 7:39.

Ramakrishnan, R., and Gehrke, J. (2000). *Database Management Systems*. New York, NY: McGraw Hill.

Ramirez-Nunez, J. A., et al. (2018). Evaluation of the detectability of electromechanical faults in induction motors via transient analysis of the stray flux. *IEEE Trans. Ind. Appl.* 54, 4324–4332. doi: 10.1109/TIA.2018.2843371

Sabbagh, R., Gawlik, B., Sreenivasan, S. V., Stothert, A., Majstorovic, V., and Djurdjanovic, D. (2020). "Big data curation for analytics within the cyber-physical manufacturing metrology model (CPM3)," in *Procedia CIRP* (Chicago, IL).

Sakamoto, Y., Ishiguro, M., and Kitagawa, G. (1986). "Akaike Information Criterion Statistics" in *Akaike Information Criterion Statistics*, ed D. Reidel (Dordrecht), 81.

Sayood, K. (2002). *Lossless* Compression *Handbook*. Amesterdam; Boston; London; New York, NY; Oxford; Paris; San Diego, CA; San Francisco, CA; Singapore; Sydney; Toky: Elsevier.

Siegel, D. (2013). *Prognostics and Health Assessment of a Multi-Regime System Using a Residual Clustering Health Monitoring Approach*. Cincinnati, OH: University of Cincinnati.

Siegel, D., and Lee, J. (2011). An auto-associative residual processing and K-means clustering approach for anemometer health assessment. *Int. J. Progn. Heal. Manag.* 2:117. Available online at: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.369.8241

Suresh, P., Thayaparan, T., Obulesu, T., and Venkataramaniah, K. (2013). Extracting micro-doppler radar signatures from rotating targets using Fourier–Bessel transform and time–frequency analysis. *IEEE Trans. Geosci. Remote Sens.* 52, 3204–3210. doi: 10.1109/TGRS.2013.2271706

Tharini, C., and Ranjan, P. V. (2009). Design of modified adaptive Huffman data compression algorithm for wireless sensor network. *J. Comput. Sci.* 5:466. doi: 10.3844/jcssp.2009.466.470

Yeap, Y. M., Geddada, N., and Ukil, A. (2018). Capacitive discharge based transient analysis with fault detection methodology in dc system. *Int. J. Electr. Power Energy Syst.* 97, 127–137. doi: 10.1016/j.ijepes.2017.10.023

Conflict of Interest: AS was employed by the company The MathWorks, Inc.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

frontiers in Computer Science

ORIGINAL RESEARCH
published: 30 October 2020
doi: 10.3389/fcomp.2020.578673

Ontology-Based Context Modeling in Physical Asset Integrity Management

Ali Al-Shdifat, Christos Emmanouilidis*†, Muhammad Khan and Andrew G. Starr*

School of Aerospace, Transport and Manufacturing, Cranfield University, Cranfield, United Kingdom

Asset management is concerned with the management practices, technologies and tools necessary to maximize the value delivered by physical engineering assets. IoT-generated data are increasingly considered as an asset and the data asset value needs to be maximized too. However, asset-generated data in practice are often collected in non-actionable form. Collected data may comprise a wide number of parameters, over long periods of time and be of significant scale. Yet they may fail to represent the range of possible scenarios of asset operation or the causal relationships between the monitored parameters, and so the size of the data collection, while adding to the complexity of the problem, does not necessarily allow direct data asset value exploitation. One way to handle data complexity is to introduce context information modeling and management, wherein data and service delivery are determined upon resolving the apparent context of a service or data request. The aim of the present paper is, therefore, 2-fold: to analyse current approaches to addressing IoT context information management, mapping how context-aware computing addresses key challenges and supports the delivery of monitoring solutions; and to develop a maintenance context ontology focused on failure analysis of mechanical components so as to drive monitoring services adaptation. The approach is demonstrated by applying the ontology on an industrially relevant physical gearbox test rig, designed to model complex misalignment cases met in manufacturing and aerospace applications.

Keywords: internet of things, context information management, maintenance ontology, context sharing, physical asset management

Edited by:
Oscar Lazaro,
Innovalia Association, Spain

Reviewed by:
Laura Belli,
University of Parma, Italy
Luca Davoli,
University of Parma, Italy

****Correspondence:***
Ali Al-Shdifat
ali.alshdifat@cranfield.ac.uk
Christos Emmanouilidis
christosem@ieee.org

†Present address:
Christos Emmanouilidis,
Faculty of Economics and Business
Operations — Management,
Groningen, Netherlands

Specialty section:
This article was submitted to
Mobile and Ubiquitous Computing,
a section of the journal
Frontiers in Computer Science

Received: *30 June 2020*
Accepted: *28 September 2020*
Published: *30 October 2020*

Citation:
Al-Shdifat A, Emmanouilidis C,
Khan M and Starr AG (2020)
Ontology-Based Context Modeling in
Physical Asset Integrity Management.
Front. Comput. Sci. 2:578673.
doi: 10.3389/fcomp.2020.578673

INTRODUCTION

Typical applications of internet of things (IoT) technologies amalgamate the ability to identify, sense, compute, communicate and sometimes actuate, for the purpose of monitoring and remotely controlling the environment (de Matos et al., 2020). According to a Statistica report. (2020), it is predicted that the amount of devices with Internet connectivity will exceed 50 billion by 2030. Such devices produce significant volumes of data which are communicated through networks, and upon processing enable better informed decision making and actions. One method of handling the high complexity of such volumes of data is by introducing context information management. Context is a key aspect in the process of leveraging information concerning situations and enabling applications to be adapted according to the perceived context (Pradeep and Krishnamoorthy, 2019).

Systems with context awareness are employed within IoT environments for the purpose of sensing the operational environment and for delivering an appropriate response to both the user and application (Perera et al., 2014). Such systems are capable of analyzing the data generated by IoT devices, generating a high-level of semantic organization of data and then converting it into context information. This information is subsequently utilized in determining an environment's status so as to drive appropriate responses. In general, the status of the environment is determined by a combination of circumstances, including users, applications, location, or devices (Abowd et al., 1999), which constitute the context information. As IoT technologies become more embedded in monitoring activities, there is a growing necessity to manage their context information in industrial environments. This entails gathering, modeling, reasoning, and disseminating context in order to efficiently manage the data generated by multiple devices and to ensure that they can be effectively integrated into enterprise systems. Nonetheless, data contributing to context information are often modeled or processed within the narrow scope of isolated subsystems, restricting interoperability. Moreover, even when similar systems for collecting context are applied in distinct settings, information is infrequently shared between them (Perera et al., 2014).

The ability to share context among different applications is a critical necessity for the IoT, making data shared between heterogeneous systems reusable in multiple applications (Ramachandran and Krishnamachari, 2019). Context information management has been recognized as a challenge for relevant research and early on Bernardos et al. (2008) developed a data fusion framework for context-awareness systems that included the following stages: (i) Obtaining context, (ii) processing context, (iii) reasoning and decision-making. Perttunen et al. (2009) have surveyed popular context reasoning and representation techniques and provided an overview of the requirements for context representation, arguing that such requirements were insufficiently covered in the literature regarding the interplay between efficiency, expressiveness, soundness, and completeness, with ontology-based approaches achieved improved scalability and reuse compared to other approaches. This finding is consistent with that of Bettini et al. (2010), although scalability of on-line reasoning with a large number of entities is raised as a significant challenge. This is the case when dealing with data of significant complexity and scale, as typically encountered in IoT applications (2020), making it important that the semantics of IoT data are captured by appropriate context modeling to gain valuable insights (Perera et al., 2014).

Context information management has largely dealt with the challenges of ubiquitous environments, as well as the data heterogeneity and services scalability. Nonetheless, while substantial research efforts have been devoted to context information management in web-based, mobile and ubiquitous computing, including IoT-enabled computing, little attention has been given to translate these advances to tangible progress in industrial monitoring services (Al-shdifat and Emmanouilidis, 2018). Context modeling in the literature is typically handled *via* ontologies. However, when dealing with monitoring services in manufacturing environments, developed approaches often lack expressiveness concerning the representation of the domain knowledge. To address such needs, this paper analyses requirements and produces a design for the components required to develop effective context-aware systems to enhance monitoring services in industrial environments. It then presents the development of a context resolution service focused on failure analysis of mechanical components so as to drive monitoring services adaptation. The paper is structured as follows. Section related work briefly discusses literature related to context information sharing and ontologies in maintenance and asset management. Section system framework and methodology presents the system framework and the ontology development process, based on established practice and maintenance vocabulary standards. An instantiation of the developed ontology is implemented for testing on an industrially relevant test rig and is presented in section implementation on a case study. Section results and discussion presents and discusses the ontology design and its implementation, including examples of context resolution results. The final section summarizes the key contributions of the paper and suggests potential further research pathways.

RELATED WORK

The following section presents a discussion on the basic concepts in the field of context-aware systems including context, context-awareness in IoT, the context information sharing, as well as the ontologies in maintenance and asset management.

Context Information Management

The concept "context-aware" system was originally proposed by Schilit and Theimer in 1994 stating that "A system is context-aware, if it uses context to provide relevant information and/or services to the user." Other early works have defined context as "any information that can be used to characterize the situation of an entity, an entity is a person, place, or object that is considered relevant to the interaction between a user and an application" (Abowd et al., 1999). Abowd and Mynatt (2000) specified the basic elements required for analyzing and understanding context, namely the five Ws (what, who, where, why, and when). According to Byun and Cheverst (2004), a system is defined as being context aware if it is capable of extracting, interpreting and using context information and its functionality can be adapted to the prevailing context of use. In the domain of asset and maintenance management, the early definition of context by Abowd et al. (1999) can be adopted and extended by specifying that context is relevant to the asset and its hierarchy, the user, the production or service business circumstances, as well as overall system and operating environment aspects (Emmanouilidis et al., 2019). Despite the generally acknowledged definitions of what is regarded as context awareness, a standard format and representation of the concept has not been established (Xu et al., 2014; Perera et al., 2015; de Matos et al., 2017). Various researchers have determined different typologies of context. Abowd et al. (1999) differentiated between primary and secondary context, in addition to conceptual and operational. Liu et al. (2011) stated that context can be classified as user, physical or networking. **Table 1** provides a summary of the different approaches adopted for categorizing context.

According to Perera et al. (2015), the steps required for a system to deliver context information are acquisition, modeling, reasoning, and distribution which combined form the context

TABLE 1 | Different context categorization schemes.

References	Context categorization
Abowd et al. (1999)	Who, Where, When What, and Why
Chen and Kotz (2000)	User, Computing, Physical, and When
Henricksen (2003)	Sensed, Static, Profiled, and Derived
van Bunningen et al. (2005)	Operational and Conceptual
Chong et al. (2007)	Computing, Physical, Historical, and Sensor
Rizou et al. (2010)	Observable and Non-Observable
Liu et al. (2011)	User, Physical, and Networking
Emmanouilidis et al. (2013)	User, Environment, System, Social, Service
Valverde-Rebaza et al. (2018)	Location and Social

lifecycle. In the acquisition step, the raw data are collected from sensors, databases or the surrounding environment. In the modeling stage, the data is brought into a particular representation so it can be converted into input for the reasoning stage. Various approaches have been described in the current literature for the modeling process, including key-value pairs, ontology, and markup schemes (Bettini et al., 2010; Snidaro et al., 2015). The semantic processing stage in the context lifecycle is the reasoning process, where different methods can be used for inferring context, such as such as supervised/unsupervised learning, rules, ontologies, probabilistic approaches, as well as data aggregation and fusion mechanisms (Perera et al., 2015). Hence, the context-awareness of a system is determined by its ability to utilize the context acquired via the context lifecycle to deliver beneficial information/services to users (Abowd et al., 1999). Several context-aware systems utilize context purely for decision support/making or direct distribution to the end user. Nevertheless, certain systems allow context information to be shared with other interested actors or subsystems. This is defined as context information sharing and represents one of key challenges in the field of context-awareness for IoT (Perera et al., 2015; Boavida et al., 2016).

Industry 4.0 and Context Interoperability in IoT

When considering IoT usage in industrial environments, the term Industrial Internet of Things (IIoT), or simply Industrial Internet, is employed, and is being considered as fundamentally linked to Industrie 4.0 (I4.0) (Jeschke et al., 2017). Comprising technologies such as IIoT, robotics/automation/control, additive manufacturing, simulation, cloud-based computing and platforms, industrial security, cognitive computing and artificial intelligence, mobility and wearables, big data and analytics, systems integration, augmented and virtual reality, as well as smart and new materials, I4.0 gives rise to new services and business models (Frank et al., 2019), driven by such technologies. Product Lifecycle Management (PLM) systems are particularly benefitting from such technologies to connect physical assets and products, processes, data, people and business systems (Keivanpour and Ait Kadi, 2019) exploiting product embedded sensor and intelligence capabilities, including product or process Condition Monitoring (CM) capabilities. Recent developments in IoT technologies have led to a renewed interest in context-aware computing. Context-awareness plays a central role in defining what data needs to be collected and how to be processed, as well as in determining what information and services are required to be made available to a consumer of data or services (Perera et al., 2014; Sezer et al., 2018). Context management is considered to comprise context acquisition, modeling, reasoning, and dissemination (Perera et al., 2014). **Table 2** summarizes surveys of IoT context-aware systems from 2010 to 2020.

Context information can be provided in various different ways, including variations in format, length, type, and representation of the data (de Matos et al., 2020). Hence, there is a need to ensure that context sharing platforms offer context interoperability. Context-relevant data can be produced by IoT entities and context management needs to be handled through a context management information processing layer. This layer would be expected to handle context data produced from multiple sources, including third-party software. Therefore, context sharing functionality is facilitated by a context sharing platform. The platform is capable of creating semantic interconnections between domains *via* the sharing of context information. As IoT environments can be highly complex, context-sharing platforms must be capable of dealing with a range of situations and implement service adaptation mechanisms driven by context building blocks (de Matos et al., 2020). These building blocks can be categorized as Properties and Architectural Components. The former applies to predominantly software aspects of context sharing platforms, including Modeling, Reasoning, Dissemination, Processing, Interoperability, Privacy, Scalability, and Availability, as shown in **Table 3**.

Architectural considerations regarding enabling hardware for the deployment of a context sharing platform, include communication technologies, storage space, and processing layers. Furthermore, some building blocks are strictly related to the context sharing properties (e.g., Modeling, Reasoning, and Dissemination), which are those that are specifically required in industrial monitoring. There are a variety of different IoT platforms, frameworks, services and middleware that are capable of collecting, processing and analyzing sensor data. In this regard, various researchers (Perera et al., 2015; Mineraud et al., 2016; Sezer et al., 2018; Pradeep and Krishnamoorthy, 2019; de Matos et al., 2020) have produced surveys of such IoT platforms, frameworks, systems, prototypes, middleware, and various different techniques and some representative examples are listed in **Table 4**, showing also their context modeling, reasoning, and dissemination features.

While all the approaches deal with some form of context management, starting from acquisition and modeling, eventually actionable context needs to be domain—specific. In the application domain of asset and maintenance management, context strongly depends on assets and their hierarchy. Unless such context is captured, it is hard to convert IoT-generated data from industrial systems to actions. Therefore, it is important to create a representation that integrates qualitative and quantitative

TABLE 2 | Summary of IoT context-aware system surveys.

Survey title	Year	Contribution	References
A survey of context modeling and reasoning techniques	2010	State-of-the-art in context modeling and reasoning in pervasive computing.	Bettini et al., 2010
Context aware computing for the internet of things: a survey	2014	Comprehensive survey and analysis of context awareness for internet of things.	Perera et al., 2014
Engineering context-aware systems and applications: a survey	2016	Context-aware systems and applications in engineering.	Alegre et al., 2016
Internet of things: a review of surveys based on context aware intelligent services	2016	A meta-survey of surveys on context awareness	Gil et al., 2016
Context-aware computing, learning and big data in internet of things: a survey	2018	Context awareness for IoT	Sezer et al., 2018
The MOM of context-aware systems: a survey	2019	Comparison of several context-aware systems	Pradeep and Krishnamoorthy, 2019
Context information sharing for the internet of things: a survey	2020	Presented essential building blocks for the development of context sharing platforms and reviewed the challenges and open issues for such platforms.	de Matos et al., 2020

data, wherein data and service delivery is determined upon resolving the apparent context of a service or data request. The most common approaches to achieve this, as seen in **Table 3**, is through ontology-based modeling. An ontology formally represents knowledge through concepts and relationships that exist in a specific domain and are a key construct of the semantic web (Gayathri and Uma, 2018).

Ontologies in Predictive Maintenance and Asset Management

As the manufacturing environment is becoming knowledge-intensive and more dynamic, maintenance is becoming more and more crucial in Asset Lifecycle Management. The use of semantic technologies, particularly ontology-based modeling for predictive maintenance, has become an important research topic and thus many ontologies have been offered to promote knowledge representation and reuse within the context of predictive maintenance. Medina-Oliva et al. (2014) developed a knowledge model for fleet predictive maintenance to handle fleet-wide contextual knowledge, arguing that fleet-wide Prognostics and Health Management (PHM) requires a knowledge-based system capable of handling contextual information. Thus, decision-making processes for diagnosis and maintenance are strengthened by semantic modeling, which deals the definition of concepts and relationships between them. In another example, an ontology was developed for the predictive maintenance in the wind energy domain and used as a basis for the identification and diagnosis of faults for monitoring the condition of wind turbines (Papadopoulos and Cipcigan, 2010). The proposed ontology model was used, by conducting ontology queries, to detect potential failures and their specific locations in the gearbox of the Wind Energy Converter (WEC).

When considering the manufacturing domain, it is of interest to capture the functional impact of asset integrity level on the actual manufacturing process. Castet et al. (2018) presented an approach for capturing fault information in a modeling environment using ontology of fault management and a set of plugins designed to automatically extract two reliability artifacts, the FMECA and fault tree. FMECA offers a sound basis upon which to express the organizational and functional association between a manufacturing asset hierarchy and its linkage with the functional integrity of the production facility. In the same year, Nuñez and Borsato (2018) conducted another study proposing an ontological model called OntoProg, serving as a widely agreed data and knowledge representation scheme for diagnostic-oriented maintenance, capable of being used in different types of industrial machines, and a set of SWRL rules to improve the ontology's expressiveness were suggested. In another recent example of an ontology-based approach to predictive maintenance, fuzzy clustering is employed to infer the criticality of failures, while SWRL rules are employed for predictive reasoning for the transition between states of different criticality, without though applying context-specific modeling an reasoning (Cao et al., 2019). Ontological approaches to support maintenance management that employ industrial scenarios have been developed for a range of assets, including urban infrastructure (Wei et al., 2020), highway infrastructure (France-Mensah and O'Brien, 2019), Building Information Management (BIM) (Farghaly et al., 2019), transport infrastructure (Ren et al., 2019; Li et al., 2020), and railway infrastructure (Dimitrova et al., 2020). **Table 5** summarizes of ontologies in maintenance and asset management.

The review of the related research work reveals two issues. Firstly, there is a missing link between knowledge constructs and context-dependent operational reliability-based services adaptation actions. Focusing on the asset context, relevant domain knowledge can be modeled in many forms but of particular interest are knowledge constructs relevant to reliability analysis, such as Fault Modes, Effects (and criticality) Analysis, FME(C)A. FMEA or FMECA models are however often utilized as a design-stage engineering study. Maintenance services, on the other hand, need to be invoked during the operating time and, thus, relevant information representations need to be enriched to enable dynamic context

TABLE 3 | Context sharing concerns.

Context sharing properties	Type	Aim	Implementation features	References
Modeling (M)	Properties	Responsible for mapping context into a predefined format.	Key-value, markup scheme, graphical, object oriented, logic-based, ontology-based, and hybrid context modeling.	Chen and Kotz, 2000; Perera et al., 2015
Reasoning (R)	Properties	Defined as the process to obtain high-level information from less enriched, or even raw data	Supervised and unsupervised learning, rules, fuzzy logic, ontology-based, probabilistic reasoning.	Bettini et al., 2010; Perera et al., 2015
Data Dissemination (D)	Properties	The context information is shared to relevant entities	Static and dynamic.	Perera et al., 2015
Privacy (P)	Properties	Data on the context includes private data, such as User ID, preferences, activities, and location. Although these drive context, privacy preservation should apply.	Access control policies, anonymization, cryptography.	Tiburski et al., 2015
Interoperability (I)	Properties	Heterogeneity of data requires that different subsystems or systems must be interoperable	Interoperability through format, source, length, and representation, and semantic alignment	de Matos et al., 2020
Context Processing (CP)	Properties	It aims to obtain, produce, and share context information to service a data or service request	Searching, filtering, and aggregation.	Lunardi et al. (2015)
Availability (AV)	Properties	The context must be always available for possible sharing	Availability ensured *via* cloud platforms or cached data	de Matos et al., 2020
Communication technologies (C)	Architectural Components	It refers to all equipment and programs that are used to process and communicate information	Communication devices, channels, and protocols for external and internal networks	Doukas et al., 2015; de Matos et al., 2020
History (Hi)	Architectural Components	Past data or inferred context stored locally or over the cloud.	Locally or cloud—based	de Matos et al., 2020
Architectural model	Architectural Components	Architecture can follow different patterns to support context sharing	Cloud-based, centralized-edge, and decentralized-edge	de Matos et al., 2020

inference and the composition of contextually relevant services. Secondly, existing predictive maintenance approaches in the manufacturing domain are still limited to the deployment of condition monitoring systems for identifying the failure mode and effects analysis in mechanical components. Therefore, the resolution of asset context is needed to analyse mechanical systems and logically connect measurements, observed behavior and intended function, with machinery operating condition and faults. To this end, FMEA offers appropriate grounding for the baseline of the knowledge mapping. According to Keivanpour and Ait Kadi (2019), failure mode analysis based on FME(C)A is recommended to ensure that maintenance activities are consistent with established fundamental practice-oriented knowledge. The following section presents an ontology-based development to describe knowledge through concepts and relationships that exist in a specific domain.

SYSTEM FRAMEWORK AND METHODOLOGY

This section presents the design of a system framework to develop a maintenance context ontology focused on failure analysis of mechanical components so as to drive monitoring services adaptation. The proposed ontology for the context resolution mechanism is relevant to failure analysis of mechanical components, and the terminology and relationships between concepts are structured on the basis of relevant standards with a reliability-oriented knowledge grounding. A mechanism for reasoning is being utilized for the delivery of context resolution and the obtained context can introduce a metadata layer on data or events produced by either automation or human-driven means. An example of 6health management of rotating machinery is utilized to offer a basis for the domain context, but the actual upper level ontology expressiveness is such that can apply to a range of machines by extending it through more specialized or application specific detailed ontologies.

The ontology is being utilized for the storage of knowledge relevant to fault diagnosis and reliability analysis through monitoring techniques. Hence, it is possible to query which type of approach for condition monitoring should be used and in what manner. Thus, queries can be made about what kind of condition monitoring technique that should be used and how. Additionally, inferences can be drawn in the

TABLE 4 | Comparison of context-awareness features of existing approaches.

Approach name	Year	Category	Modeling	Reasoning	Dissemination	References
Context Toolkit	2001	Toolkit	Key-value	(✓) Provided but not mentioned	Query	Dey et al., 2001
Aura	2002	Middleware	Markup Schemes	Rules	Publish	Garlan et al., 2002
CoBrA	2004	Middleware	Ontology-Based	Rules, ontology-based	Query	Chen et al., 2004
CARS	2005	System	Key-Value	Un-Supervised	(✓) Provided but not mentioned	Wilson et al., 2005
MoCA	2007	Middleware	Markup Schemes, Ontology-Based	Ontology-Based	Publish, Query	de Rocha and Endler, 2006
CoSM	2009	Model	Ontology-Based	Ontology-Based	Dynamic	Yamamoto et al., 2009
ConCon	2014	Middleware	Key-Value	Ontology-Based	Static	Madhukalya and Kumar, 2014
RCOS	2016	Middleware	Ontology-Based	Ontology-Based	Dynamic	Dhallenne et al., 2016
PSW	2017	Model	Ontology-Based	Ontology-Based, Rules	Dynamic	Ruta et al., 2017
CoaaS	2018	Middleware	(✓) Provided but not mentioned	Rules, Pro	Dynamic	Hassani et al., 2018
SCENTS	2019	Middleware	(✓) Provided but not mentioned	Rules	Dynamic	Liu et al., 2019

TABLE 5 | Summary of ontologies in maintenance and asset management.

Survey Title	Ontology domain	Contribution	References
A formal ontology for semantics in maintenance platforms	Production system	An ontology to produce new knowledge in the field of industrial maintenance that supports decision-making in the maintenance process.	Karray et al., 2012
Ontology-based implementation of an advanced method for time treatment in asset lifecycle management	Lathe machine	Implemented a method for exploiting the characteristics of time in maintenance asset lifecycle management (ALM) systems.	Matsokis and Kiritsis, 2012
Ontology-based schema to support maintenance knowledge representation with a case study of a pneumatic valve	Pneumatic valve	A methodology for knowledge representation using ontology concepts is proposed to overcome the problems of heterogeneity and inconsistency in maintenance records.	Ebrahimipour and Yacout, 2015
A novel maintenance system for equipment serviceability improvement	Manufacturing machine	A maintenance system for real-time equipment that integrates augmented reality (AR) for context-aware overlay of textual and graphical maintenance instructions on the maintenance scene.	Ong and Zhu, 2013
Context-aware recommendation for industrial alarm system	A power generation plant	An industrial alarm management system through semantic web technology and machine learning techniques.	da Silva et al., 2018
Semantic data model for operation and maintenance of the engineering asset	N/A	Proposed a semantic data model for engineering asset management, focusing on the operation and maintenance phase of its life cycle.	Koukias et al., 2013
Context modeling with situation rules for industrial maintenance	knowledge gateway system	A knowledge modeling approach and a technical architecture of a gateway system developed to support maintenance personnel.	Aarnio et al., 2016
A research on intelligent fault diagnosis of wind turbines based on ontology and FMECA	Wind turbine	A method of intelligent wind turbine fault diagnosis based on ontology and FMECA is proposed.	Zhou et al., 2015
Building an ontological knowledgebase for bridge maintenance	Transport infrastructure	"An ontology to achieve automatic rule checking and improve the management and communication of knowledge related to bridge maintenance."	Ren et al., 2019

sense that it is possible to make a comparison between an obtained value and specific thresholds based on relevant ISO standards in order to determine whether the value can be categorized as Good, Satisfactory, Alert or Alarm. Therefore, if the recorded value is considered to be in the Alert category, the system diagnoses that a failure could occur and a maintenance notification is issued for the machine indicating that intervention is required. Subsequent to the identification of an alert notification, it is then necessary to connect it with diagnostic information of the mechanical part being investigated, which will allow the failure mode and the potential causes to be determined.

Nonetheless, such simple threshold-based rules often fail to apply in practice and in view of that the ontological approach does not seek to replace actual diagnostics techniques, which may involve far more efficient and sophisticated data processing. Instead it acts as a meta-layer of knowledge to drive services adaptation, and as such could work in synergy with other techniques of data processing and condition monitoring approaches. The intended end result is that the proposed maintenance intervention is more directed, and tailored to the apparent context of a situation. The process was applied as follows:

- A concept knowledge base is established and created on the basis of the professional expertise of mechanical engineers. The fundamental concept knowledge pertaining to the domain of condition monitoring is founded on professional expertise, associated studies and standard specifications. Extraction and representation of the necessary signal functions is then performed.
- The knowledge is then transformed into ontology and SWRL rules. The ontology editor Protégé 5.5 along with its plugins (e.g., SWRL editor) are utilized in this stage.
- Implicit knowledge is extracted from the knowledge base by the ontology management system according to the SWRL rule engine.
- Identification of the source of the vibration is performed using the obtained signal as input and then conducting Pellet reasoning. **Figure 1** provides more details about the system.

Maintenance Ontology Development

The development of ontology can be based on one of the numerous procedures described in the literature, including Uschold and King, Grüninger and Fox, Methodology, Ontology Development 101 (Noy and McGuinness, 2001) (OD1) and KACTUS. The OD1 is utilized in this study as it is broadly employed (Gong and Janssen, 2013; Lau et al., 2014; Nuñez and Borsato, 2018; Ren et al., 2019), has been demonstrated to be highly appropriate for maintenance modeling, and has been extensively documented for application in the Protégé environment.

The OD1 initial step is to determine the scope and domain. In this phase, the focus of the maintenance ontology is on modeling failure analysis of mechanical components to answer queries regarding how faults manifest themselves and how they can be prevented or addressed, so as to adapt relevant diagnostics or maintenance actions in a Condition-Based Maintenance setting. Therefore, the domain of the model is Maintenance.

The next OD1 steps are to consider reuse and enumerate terms. In this regard, ontological models developed by other researchers should be considered to determine their adaptability to the current research proposal such as those proposed in Nuñez and Borsato (2018), Sanislav and Miclea (2015). Moreover, this phase involves the enumeration of all terms pertinent to the area of the ontology being developed. Therefore, the main terminology and the associated definitions are based on consolidated academic literature and mostly on established international standards, such as condition monitoring, diagnostics and maintenance (ISO, 2012, 2017), vibration analysis (ISO, 2018), failure analysis (IEC 6030 0–1, IEC 6030 0-3-1), monitoring parameters: (ISO, 2011), asset management (ISO, 2014), and MIMOSA (www.mimosa.org) standards.

In the next phase, all classes and sub-classes are classified following a top-down approach. It starts with the most general definition of a domain concept and then continues with the more specialized concepts. In this context, the main class is of the Maintenance Ontology includes the subclasses Asset, FMEA_Technique and ConditionMonitoringParameters. Every such class has its own subclasses for example, subclass FMEA_Technique has subclasses: FailureEffect, FailureMode, PotentialCause, and Symptom. An example of class hierarchy is shown in **Figure 2A**.

The next steps in this process include the design of entities and properties. Entities are all the subjects of the studied domain; properties are verbs that clarify the relations between subjects and objects, or between-subjects themselves. Subsequent to defining the class hierarchy, it is important to determine the class relationships. They need to be accompanied by three distinct types of properties: object properties, data properties. The object attribute explains the associations among distinct classes. The data property explains the properties of certain occurrences both quantitatively and quantitatively. **Figures 2B,C** show the aforementioned object properties, data properties. The final stage is to create specific individual class instances within the class hierarchy, which involves: (1) selection of the class, (2) creation of an individual occurrence of the class, and (3) filling slot values. These instances are used in the representation of particular elements. **Figure 2D** presents class individuals.

Along with identification of the procedure that has been adopted, the development of ontology models requires tools that can support all activities in the development process. Such tools include TopBraid and OntoStudio, as well as open ones, such as the popular OntoEdit, HOZO and Protégé. Specifically, Protégé is the most dominant ontology publisher due to the fact that it is an open platform that offers plug-in extensibility as well as XML (S), OWL, RDF (S), and Excel support, along with graphic taxonomy, queries in SPARQL, rules in SWRL language, and a reasoner (Pellet). The combination of OWL/SWRL provides a more flexible ontology language for modeling knowledge domains with a greater degree of expressiveness than using OWL alone (Lawan and Rakib, 2019). The SWRL is a W3C recommendation that extends horn-clause rules to OWL. OWL has demonstrated significant expressive powers over other ontology languages as the recommended ontology language for the semantic web. While OWL ontologies provide simple, reusable and easy to understand domain knowledge models, they lack the declarative expressiveness offered by rules developed in SWRL.

IMPLEMENTATION ON A CASE STUDY

The applicability of the developed ontology model is shown by utilizing a real physical asset. Gearboxes have broad utilization in numerous applications such as machine tools, industrial devices, conveyors and essentially any form of rotatory power transmission equipment in which the torque

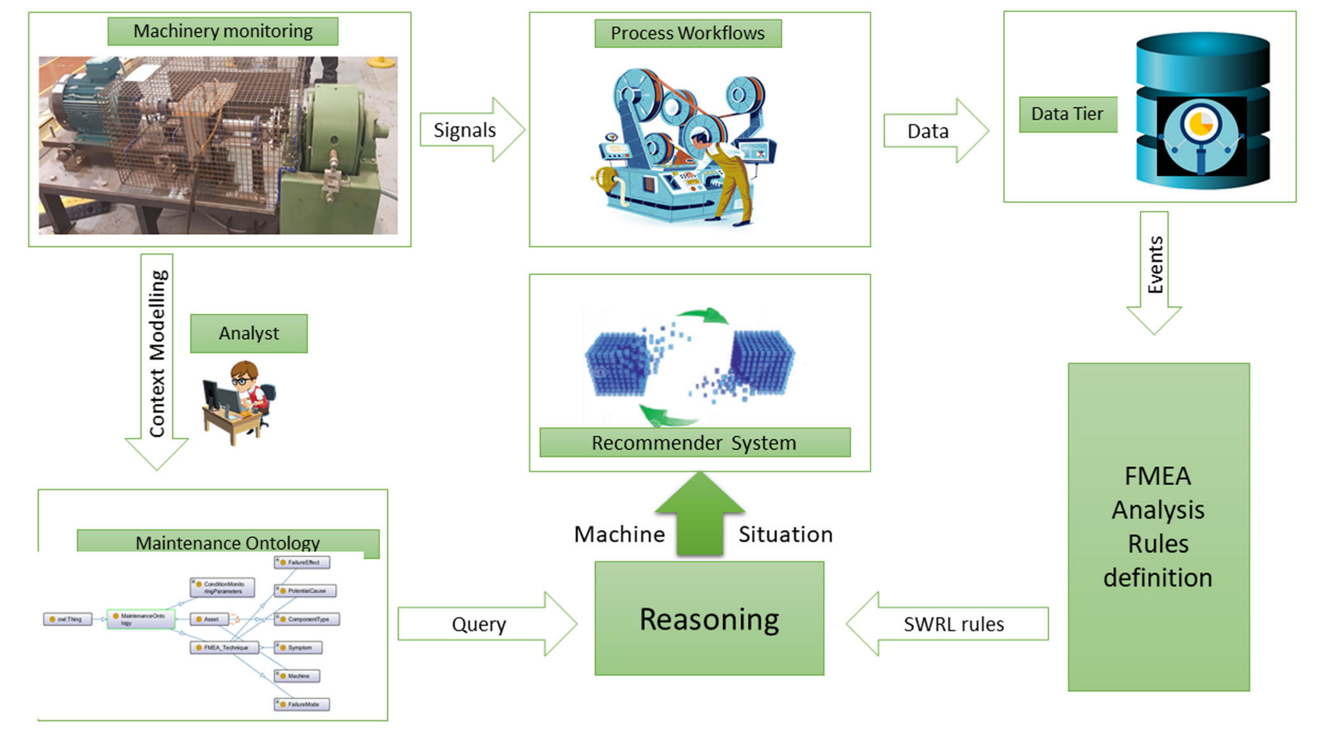


FIGURE 1 | Prototype framework.

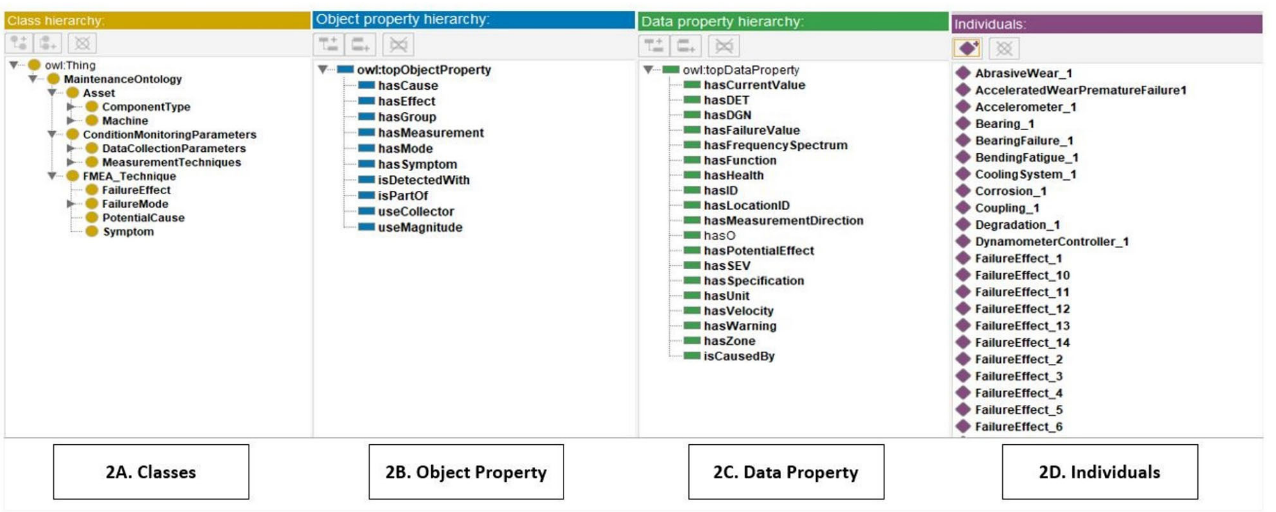


FIGURE 2 | Ontology classes, object property, data property, and individuals. **(A)** Classes. **(B)** Object property. **(C)** Data property. **(D)** Individuals.

and speed requirements need to be changed. If such devices fail, the results can often be catastrophic accidents with serious consequences. Therefore, a proactive approach must be adopted that enables such components to be monitored in real-time utilizing predictive maintenance methods (Khan et al., 2019). In the present study, the technique of vibration analysis is approached. Techniques used for assessing the health of components based on vibration are regarded as applicable for numerous reciprocating and rotating machines (Giurgiutiu et al., 2001; Bajrić et al., 2011). A laboratory-based test rig was employed and data was collected from its operation, and maintenance records. This has been designed for emulating

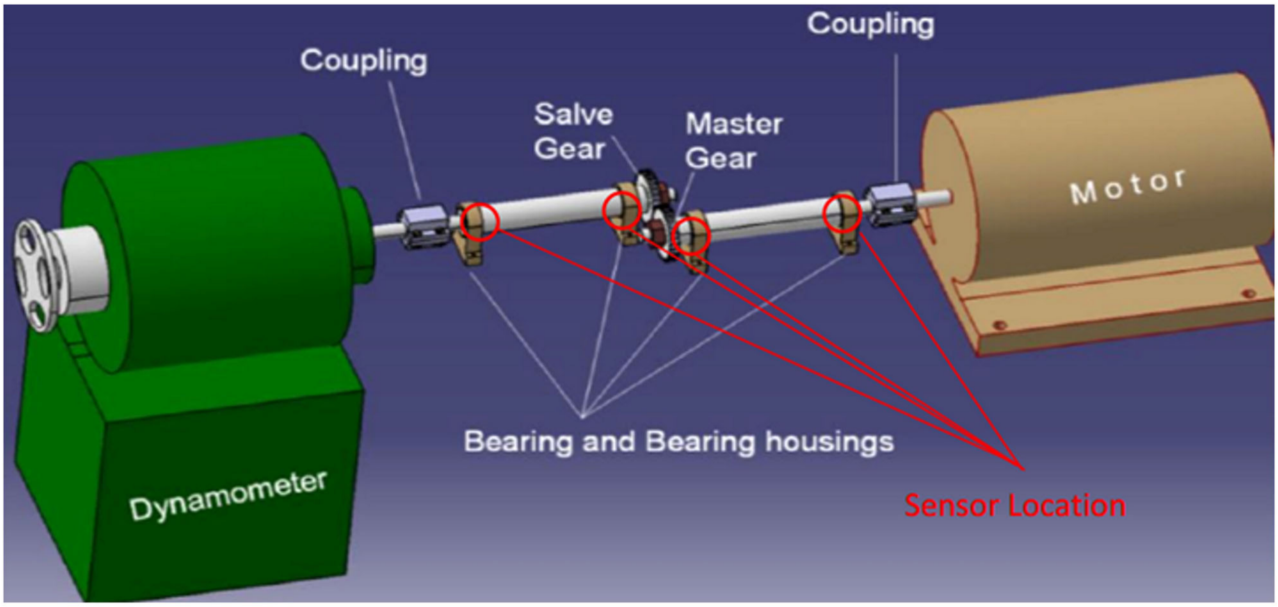


FIGURE 3 | CAD rendering of drive system and bearing locations.

complex cases of misalignment, relevant to manufacturing and aerospace engineering assets.

In order to capture the operational health of the machine, the test rig must be analyzed, allowing for an understanding on how to best capture the degradation effect on the test machine. As stated in ISO/FDIS 17359:2002(E), which details the flowchart for starting the condition monitoring process, the start is to choose the machine components in the maintenance ontology. Then, the necessary functionality of each of the components is explained for the machine to operate correctly. Additionally, all failure modes, effects, causes, symptoms and measurement approaches pertaining to the machine components are inputted utilizing the FMEA method. Subsequently, the implementation of the FMEA method indicates the most suitable measurement locations and their limits for the measurement of values by employing the vibration analysis method for prediction, which are based on ISO (2002), ISO (2016), and ISO (2009). The most pertinent components along with the most appropriate measurement methods are identified by utilizing the FMEA classification, which assigns weights according to the highest severity (SEV), occurrence (OCC), detectability (DET), and risk priority number (RPN).

Failure Mode and Effect Analysis

Asset context must be resolved for the analysis of mechanical systems and to establish logical connections between measurements, perceived behavior and the desired functionality, and the operating health and defects. In this regard, Fault Modes and Effects Analysis (FMEA) provides a suitable basis for the baseline of the knowledge mapping (IEC, 2018) due to various reasons. First, the qualitive components renders it suitable for the abstraction of maintenance knowledge focused on reliability. Second, the quantitative component allows maintenance tasks to be prioritized on the basis of measurements conducive to an approach based on risk. Third, its bottom-up structure allows failure to be assessed starting from the basic level of production systems; in other words, data are analyzed from machinery parts through to the overall system. The initial stage involves the determination of the specific aspects of the machine that have the potential to fail and then to comprehend the causes and effects of such failures (FMEA). Based on the study, it will become apparent where the data will be most accurate in highlighting the degradation of the machine being tested. Alongside the misalignment testing carried out on the machine, the rig can test the effects of loading through the dynamometer and the subsequent effects that loading will have on the system (**Figure 3**). Based on the FMEA **Table 6** (del Castillo et al., 2020), the most frequent outcome of misalignment of the gearbox will be vibration and power transfer loss through gearbox, as revealed by the RPN values. Subsequently, the vibration is spread across the machine and is most pronounced in specific locations, namely the bearings, and it is possible to easily capture the transfer loss by calculating the RPN difference between the driving motor and the loading dynamometer. As determined by the FMEA analysis, the two potential failures that are identified as having the greatest level of severity if misalignment or loading occur in the system are degradation of the gear teeth in the gearbox (RPN 150) as well as the bearing degradation (RPN 140).

Wearing of the teeth is generally caused by misaligned gears, excessive loading and lastly, a lack of lubrication. Degradation of the bearings is caused by wearing of the teeth in the gears as well as the impact of gear vibration being transferred to the shaft and then to the bearings. If the shaft of the gear is short

TABLE 6 | FMEA of test rig.

Item (ID)	Function (requirements)	Failure mode	Failure effects	SEV	Failure causes	OCC	Mitigation	DET	RPN
Bearing	"To achieve a smooth, low-friction rotary motion or sliding action between two surfaces"	Abrasive wear	Reduce fatigue life and misalignment in the bearing	6	lubricant condition, grease degradation, and improper isolation	4	Lubricant inspection and proper isolation, Monitor Shaft alignment	4	96
		Bearing seizure	Crack formation on rings and balls or rollers—Skidding	4	Inadequate heat removal capability—Loss of lubricant—High temperature—Excessive speed	3		3	36
		Noisy bearing	Surface fatigue—Glazing—Microspalling of stressed surfaces	4	Loss of lubricant—Housing bore out of round—Corrosive agents—Distorted bearing seals	3		2	24
		Fatigue (Spalling)	Bearing failure	3	Excessive loading (cyclic), misalignment	5		1	15
		Vibration	Scuffing—Fretting—Pitting of surfaces	7	Misalignment—Housing bore out of round—Unbalanced/excessive load	4		5	**140**
Gear	"To transmit shaft power on predetermined or designed angular velocities"	Tooth wear	Loss of rotation transfer, eventual gear vibration, noise	6	Contaminants in the gear mesh area or lubrication system	5	"Lubricant inspection, Regular inspection surface sanding"	5	**150**
		Scuffing	Wear and eventual tooth failure	5	Lubrication breakdown	2		4	40
		Tooth shear	Fracture	6	Tooth failure	2		3	36
		Spalling	Mating surface deterioration, welding, galling, eventual tooth failure	4	Fatigue	1		2	8
		Root filet cracking; Tooth end cracks	Surface contact fatigue and tooth failure	5	Tooth bending fatigue	2		2	20
		Pitting	Tooth surface damage	6	Cyclic contact stress transmitted through lubrication film	2		2	24

The bold values refer to components of high significance (RPN) in failure analysis.

and hard, and the bearings are situated in close proximity to the center where meshing of the gears occurs, this can be a source of vibration, which can be measured by placing sensors at the bearings as shown in **Figure 3**.

RESULTS AND DISCUSSION

The aim of ontology-based context modeling is to produce a semantic organization of data so as to drive maintenance services adaptation. When users interact with systems in this regard, the proposed maintenance ontology can help them (e.g., maintenance engineers) to narrow down the list of options by providing answers to questions such as:

- What are the common failures and diagnostic approaches for a given machine type?
- Which physical parameters to measure/use?
- What is the recommended preventive or corrective action for a specific failure mode of an asset?

An example of a typical utilization scenario is that during condition monitoring queries could be raised to resolve the monitoring service context. For instance, this could be related to determining the failure modes of a part, the functional effect of a defect on the operation of the test rig, the measurement alternatives suitable for specific defects and parts, in addition to the relevant measurement parameters. In the context of the present study, SPARQL queries were designed for the resolution of these queries. SPARQL additionally allows the federation of queries across different sources of data. By applying the following query, it is possible to determine "what are the main components of the Test rig?"

Figure 4 shows the results of a query to identify the main components of an asset type. These components are bearing,

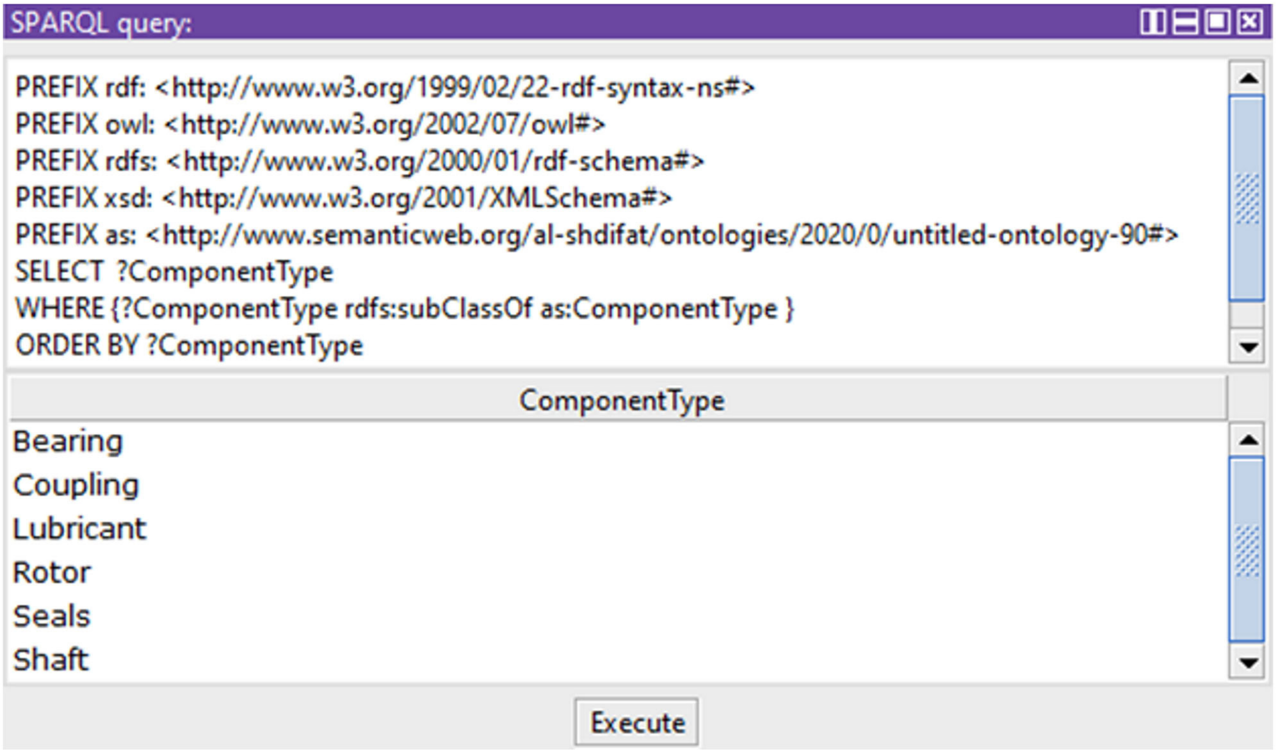


FIGURE 4 | Query result to identify the main components.

coupling, lubricant, rotor, seals, and shaft. Moreover, the present implementation allows a query in the maintenance ontology to resolve key analysis characteristics, such as components function, failure modes, causes, and effects. This query can be useful to a maintenance engineer in order to link faults to functional impacts, and other information to ensure the correct identification of the component being analyzed as shown in **Table 7**.

Another query can be applied based on FMEA to answer what are the common gearbox problems and diagnosis methods. For example, problems that arise in relation to gearboxes are misalignment, lubrication issues, bearing problems, gear teeth defects, thermal instability, and torsional and lateral vibration. Considering that the test rig under study was designed to study complex cases of misalignment in industrial machinery, the focus here will be on misalignment cases. Misalignment in gearbox arrangements can cause gear and bearing pitting, which eventually leads to complete failure. It may lead to vibrations and excessive loads that harm functioning components of the machine, such as bearings and oil seals. It is therefore important to detect and fix such issues to avoid incurring any unnecessary costs. As shown in **Figure 5**, this can take four forms: Axial misalignment; Offset or Parallel misalignment, when the centers of shafts are on different lines; Angular misalignment, when a motor shaft is at an angle to a driven component shaft; and Combination misalignment, when both angular misalignment and parallel misalignment occur.

After identifying the common gearbox problems, then it is important to identify parameters to be measured for fault detection. The developed ontology links physical measurement entities with appropriate measurement techniques. This allows to associate common faults with the physical asset and to match them with parameters or techniques appropriate for detecting the occurrence of such faults. For example, a component that has high significance in failure analysis is the bearing (**Table 8**). A critical failure mode is gear tooth wear and the typical failure effects for this is partial tooth contact (Misalignment). Another query can be applied to determine failure modes, failure causes, failure effects, symptoms, and fault severity (SEV), but also to determine the faults with highest diagnostic potential (DGN) or faults which pose the highest impact risk. SEV and DGN scale from 1 to 10, with the higher number representing the higher seriousness or risk and an appropriate query can return the faults with the highest DGN (**Table 8**) or risk. Therefore, parameters such as SEV, DGN and DET from the FMEA technique can be used within the ontology model to enable queries which in turn can identify components or processes of priority for maintenance actions.

Real data collection from the shop floor or simulated data (with "hasCurrentValue" data property) can be used to infer the component's health status and trigger alerts for decision making, such as the prognosis of a failure and the scheduling of Condition Based Maintenance (CBM) actions. In this regard, the SWRL language is being used in object properties to construct

TABLE 7 | Function, failure mode, failure cause, and failure effect for test rig.

Component	Function	Failure mode	Failure cause	Failure effect
"Shaft"	"It has the ability to translate in its axial direction, thereby changing the gear"	Misalignment1	"Angular misalignment of shaft due to mounting incorrect"	Abnormal temperature rise and excessive loading
		Corrosion	"Bearings exposed to corrosive environment"	Increased vibration and noise
"Lubricant"	"Lubricating the teeth and bearing / removing heat generated from operations"	Lubricant Degradation1	"Loss of lubricant, contaminated lubricant, aged lubricant, lubricant system failure, blocked lubrication filters, leakage"	Components failure and environmental pollution
"Bearing"	"To achieve a smooth, low-friction rotary motion or sliding action between two surfaces"	Fatigue	"Fatigue in rolling bearing parts by housing misalignment"	bearing failure
		Wear_1	"Lubricant condition, grease degradation, and improper isolation"	Sound_1
		Vibration	"misalignment"	Crack propagation
"Motor"	"Motors convert electrical energy into mechanical energy"	Overheating	"Cooling system failure, Temperature above limit, Temperature sensor failure."	UnableOperateMachine_1
		Shaft failure	"Overloading, fatigue, misalignment"	Halt generator operation and Increased vibration
		Bearing failure	"Bearing fatigue, Improper lubrication, lubricant contamination from dirt, abnormal vibration."	Increased vibration and noise

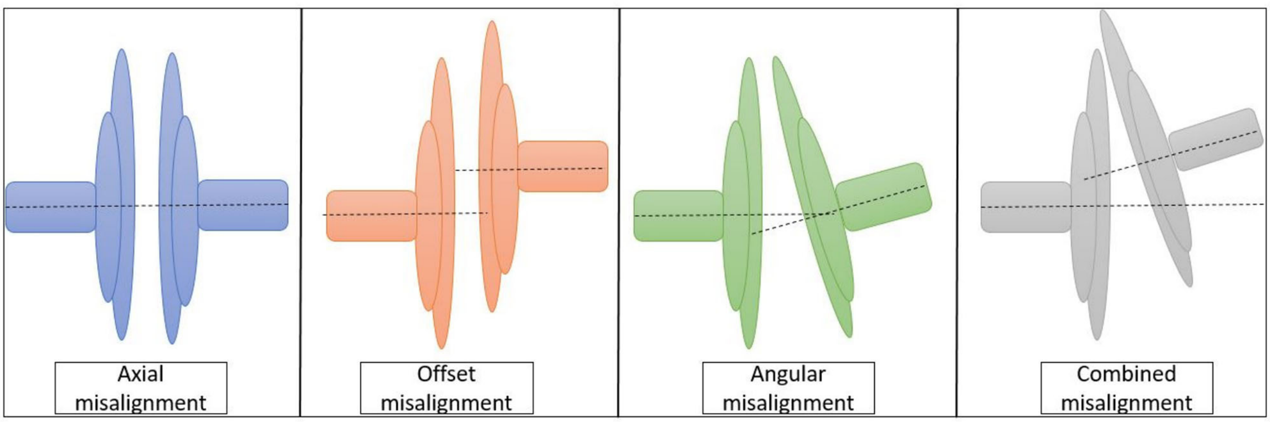


FIGURE 5 | Types of shaft misalignment adapted from Khan et al. (2019).

transitive rules (Nuñez and Borsato, 2018) and new connections are applied to the classes that allow assertion inferences to be improved. In this way the ontological approach becomes scalable: specifically SWRL built-ins (SWRLb) allow further extensions within a taxonomy. This greatly enhances the model by allowing multiple arguments according to specific real-world requirements, enabling greater expressiveness of OWL 2 languages. A transitive property is considered in cases such as: if subclass Component Type (C1) has object property has Mode, and subclass Failure Mode (FM) has object property has Cause (CA) related to subclass Potential Cause (PC), then subclass Component Type (C1) has the object property has Cause (CA) related to subclass Potential Cause (PC). Then the SWRL rule is: has Mode (?C1, ?FM1) Failure Mode (?FM1) Component (C1) Potential Cause (?PC1) has Cause (?FM1, ?PC1)—> has Cause (?C1, ?PC1). As part of the SWRL rules within the suggested ontology, (ISO, 2009) is utilized for the evaluation of the data gathered from the vibration measurements and analysis. The test rig employed in the pilot example is regarded as a mid-level asset in an asset hierarchy that includes a rolling-element bearing

TABLE 8 | Query outcome for failure mode with highest DGN.

Component type	Failure mode	Failure effect	Failure cause	Symptom	SEV	DGN	Technique
Rolling bearing	Tooth wear	Partial tooth contact (Misalignment)	Tooth failure	Vibration 3	4	5	Vibration analysis

(lower level in asset hierarch) that includes an accelerometer acting as a transducer in the data collection process. The resulting assessed parameters can be for example the velocity of the vibration in RMS mm/s, with the measurement sites as defined by standard MIMOSA VB-00, while the operating zone limits based on the (ISO, 2009) standard. To determine the machine's health status and recommended actions, the following main SWRL rules are set:

- **Component Health** (?CH1, ?E), equal(?E, "**Good**"), Measurement Location(?M1) -> has Warning(?CH1, "**Collect new data in 1 month**").
- **Component Health** (?CH 1, ?F), Measurement Location(?M1), equal(?F, "**Satisfactory**") -> has Warning(?CH1, "**Collect new data in 2 weeks**").
- **Component Health** (?CH 1, ?H), Measurement Location(?M1), equal(?H, "**Alert**") -> has Warning(?CH1, "**Schedule Condition-based Maintenance**").
- **Component Health** (?CH 1, ?I), Measurement Location(?M1), equal(?I, "**Alarm**") -> has Warning(?CH1, "**Turn off the machine**").
- **Has Health** (?CH1, ?E), equal(?E, "**Good**"), Measurement Location(?M1) -> **is Caused By** (?CH1, "**No Misalignment** ").
- **Has Health** (?CH1, ?F), Measurement Location(?M1), equal(?F, "**Satisfactory**") -> **is Caused By** (?CH1, "**Loss of lubricant - Housing bore out of round - Corrosive agents—Distorted bearing seals**").
- **Has Health** (?CH1, ?H), Measurement Location(?M1), equal(?H, "**Alert**") -> **is Caused By** (?CH1, "**Excessive loading (cyclic), misalignment** ").
- **Has Health** (?CH1, ?I), Measurement Location(?M1), equal(?I, "**Alarm**") -> **is Caused By** (?CH1, "**Misalignment—Housing bore out of round—Unbalanced/excessive load**").

Let's assume that when the data for the rolling bearing part exhibits an RMS mm/s value between zero and $\leq$2.3, then it should display a "good" notification. When values that exceed 2.3 but are below 4.5 are detected, it should display a "satisfactory" notification; a value between 4.5 and 7.1 should trigger an "alert" notification, and values in excess of 7.1 should cause an "alarm" notification that will instantly terminate the machine operation. Given this, let's assume that a value of 4.7 mm/s RMS is recorded. This is fed through the ontology, activating the Pellet plugin reasoner in the Protégé ontology editor. The outcome will be that the component's health will be inferred to be set as "Alert," producing a recommendation to "Schedule Condition-based Maintenance. Moreover, once an ALERT warning has been issued, it is then important to associate it with the diagnostic information of the analyzed mechanical component, associating the identified the failure mode with potential causes (**Figure 6**). In this way, the maintenance intervention becomes context-depended and is therefore more focused and relevant to the identified context of the monitoring situation.

While simple single-parameter threshold-based rules might be easy to interpret, they do not often hold in practice. Instead, more complex multi-parameter rules are more likely to apply. The reasoning process can replace simple rules with the activation of more complex decision functions which may be produced as a result of machine learning over monitoring historical data. The value of the described process is that it sits at a higher level of abstraction and can therefore work with different lower level computational rules.

CONCLUSION AND FURTHER RESEARCH

This paper presented the development of a context resolution service mechanism for industrial diagnostics, based on the design of a maintenance ontology focused on modeling and reasoning failure analysis of mechanical components. The maintenance ontology has been developed employing established methodologies and upon consulting a range of domain-relevant international standards. The ontology development was further applied on a physical mechanical transmission test rig. Thus, queries could be raised in terms of the resolution of the monitoring service context to determining the failure mode and its potential causes of the test rig, in addition to the relevant measurement parameters. Moreover, SWRL reasoning rules were used based on (ISO, 2009) for the evaluation of the data gathered, the prognosis of failure is being performed, sending a maintenance message for intervention in the machine. In this way, the maintenance intervention is more directed, ceasing to be exploratory. This highlighted the need for handling the whole context information management lifecycle and ontologies in maintenance and asset management to maximize the value delivered by physical engineering assets.

The outcomes of the work can be used in other industrially relevant application scenarios to drive maintenance service adaptation. While the application focus is quite specific, the ontology abstraction level is actually such that it could also be implemented on other application cases, as it offers a sound baseline for further customization or extensions. When serving different application scenarios, the derived abstract model developed with the process described in section system framework and methodology still holds, as the employed terms and relationships were developed employed established standards. However, after going through a similar process for deriving the application-specific part of the ontology as described in section implementation on a case study and the developed queries, as described in section results and discussion, additional needs can be identified, which may require

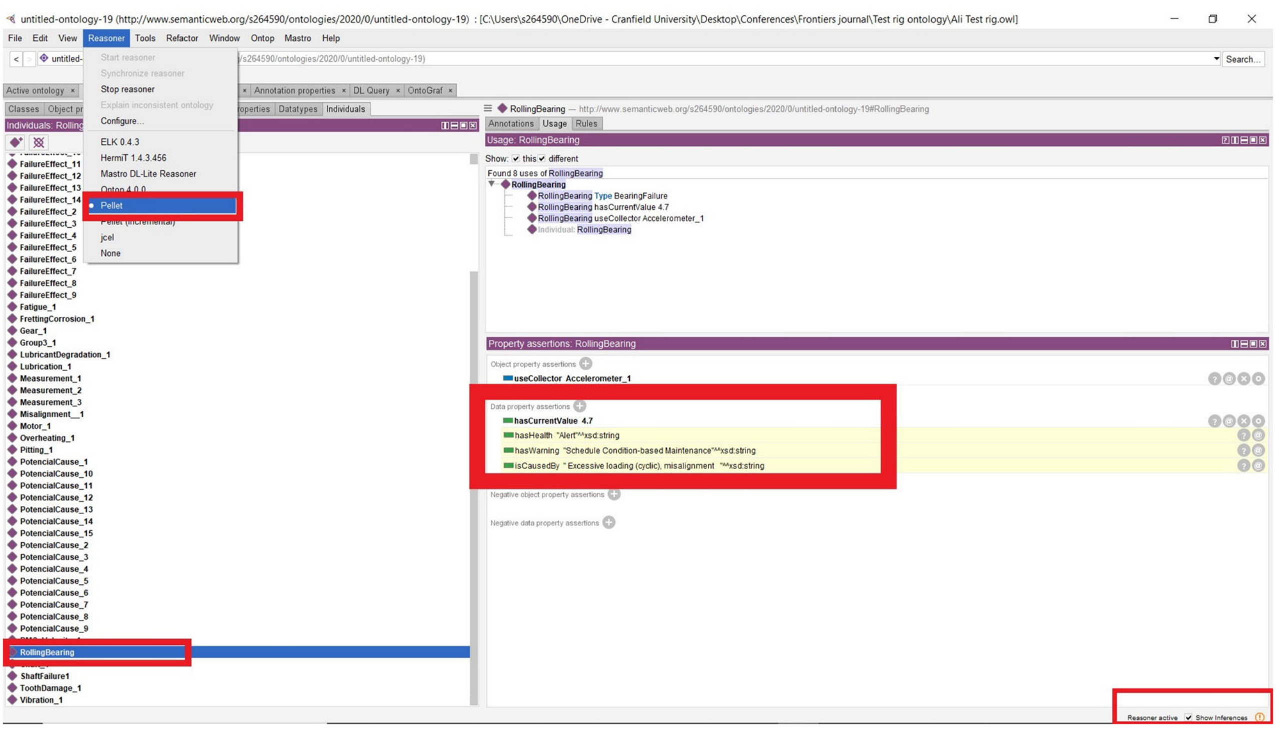


FIGURE 6 | SWRL rules for generating warning and potential causes.

the inclusion of additional entities, relationships, and queries development. This will be determined by going through an ontology assessment and evaluation cycle in the context of the new application scenario, especially regarding the ontology expressiveness and coverage. Nonetheless, this will affect largely lower abstraction level terms, rather than upper hierarchy classes and associated object or data properties. For example asset types and their associated terms if need be may be complemented by additional asset types. The higher level classes, object properties, and data properties will retain the structure of **Figures 2A–C** but the population of lower tier terms and individuals for such class structures will need to be developed for the additional asset types, as typically holds in managing ontologies. However, the example reasoning rules presented in Section results and discussion can be re-used but can be extended with additional ones to cover the coverage and expressiveness of the updated ontology.

Consequently, further research should be carried out to link the current ontology implementation with a live condition monitoring service, as well as to apply it to real industrial environments as an enabler of more efficient IoT-enabled monitoring services.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author/s.

AUTHOR CONTRIBUTIONS

AA-S and CE: original research, methodology, and paper editing. MK and AS: paper review and editing. All authors contributed to the article and approved the submitted version.

REFERENCES

Aarnio, P., Vyatkin, V., and Hastbacka, D. (2016). "Context modeling with situation rules for industrial maintenance," in *IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA)*. (Berlin), 1–9. doi: 10.1109/ETFA.2016.7733539

Abowd, G. D., Dey, A. K., Brown, P. J., Davies, N., Smith, M., and Steggles, P. (1999). "Towards a better understanding of context and context-awareness," in *Proceeeding of 1st International Symposium on Handheld and Ubiquitous Computing, ser. HUC '99*. (London, UK: Springer-Verlag), 304–307. doi: 10.1007/3-540-48157-5_29

Abowd, G. D., and Mynatt, E. D. (2000). Charting past, present, and future research in ubiquitous computing. *ACM Trans. Comput. Hum. Interact.* 7, 29–58. doi: 10.1145/344949.344988

Alegre, U., Augusto, J. C., and Clark, T. (2016). Engineering context-aware systems and applications: a survey. *J. Syst. Softw.* 117, 55–83. doi: 10.1016/j.jss.2016.02.010

Al-shdifat, A., and Emmanouilidis, C. (2018). Development of a context-aware framework for the integration of internet of things and cloud computing for remote monitoring services. *Procedia Manuf.* 16, 31–38. doi: 10.1016/j.promfg.2018.10.155

Bajrić, R., Sprečić, D., and Zuber, N. (2011). Review of vibration signal processing techniques towards gear pairs damage identification. *Int. J. Eng. Technol.* 11, 124–128.

Bernardos, A. M., Tarrío, P., and Casar, J. R. (2008). "A data fusion framework for context-aware mobile services," in *Proceedings of the IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, (South Korea: Seoul), 606–613. doi: 10.1109/MFI.2008.4648011

Bettini, C., Brdiczka, O., Henricksen, K., Indulska, J., Nicklas, D., Ranganathan, A., et al. (2010). A survey of context modelling and reasoning techniques. *Pervasive Mobile Comput.* 6, 161–180. doi: 10.1016/j.pmcj.2009.06.002

Boavida, F., Kliem, A., Renner, T., Jukka Riekki, J., Jouvray, C., Jacovi, M., et al. (2016). "People-centric internet of things–challenges, approach, and enabling technologies" in *Intelligent Distributed Computing IX—Symposium on Intelligent Distributed Computing (IDC'2015)* (Cham: Springer International Publishing), 463–474. doi: 10.1007/978-3-319-25017-5_44

Byun, H. E., and Cheverst, K. (2004) Utilizing context history to provide dynamic adaptations. *Appl. Artif. Intel.* 18, 533–548. doi: 10.1080/08839510490462894

Cao, Q., Samet, A., Zanni-Merk, C., De Beuvron, F. D. B., and Reich, C. (2019). An ontology-based approach for failure classification in predictive maintenance using fuzzy c-means and SWRL rules. *Procedia Comput. Sci.* 159, 630–639. doi: 10.1016/j.procs.2019.09.218

Castet, J. F., Bareh, M., Nunes, J., Okon, S., Garner, L., Chacko, E., et al. (2018). "Failure analysis and products in a model-based environment," in *IEEE Aerospace Conference Proceedings (IEEE)* (Big Sky, MT), 1–13. doi: 10.1109/AERO.2018.8396736

Chen, G., and Kotz, D. (2000). *A Survey of Context-Aware Mobile Computing Research.* Technical Report. Hanover, NH: Department of Computer Science, Dartmouth College, 1–16. Available online at: http://www.cs.dartmouth.edu/reports/TR2000-381.pdf (accessed June 16, 2020).

Chen, H., Finin, T., Joshi, A., Kagal, L., Perich, F., and Chakraborty, D. (2004). Meet the semantic web in smart spaces. *IEEE Internet Comput.* 8, 69–79. doi: 10.1109/MIC.2004.66

Chong, S. K., McCauley, I., Loke, S. W., and Krishnaswamy, S. (2007). "Context-aware sensors and data muling," in *Context Awareness for Self-Managing Systems (Devices, Applications and Networks) Proceeding* (Berlin: VDE-Verlag), 103–117.

da Silva, M. J., Pereira, C. E., and Götz, M. (2018). Context-aware recommendation for industrial alarm system. *IFAC-PapersOnLine* 51, 229–234. doi: 10.1016/j.ifacol.2018.08.266

de Matos, E., Amaral, L. A., and Hessel, F. (2017). *Context-Aware Systems: Technologies and Challenges in Internet of Everything Environments.* Cham: Springer International Publishing, 1–25. doi: 10.1007/978-3-319-50758-3_1

de Matos, E., Tiburski, R. T., Moratelli, C. R., Johann Filho, S., Amaral, L. A., Ramachandran, G., et al. (2020). Context information sharing for the internet of things: a survey. *Comput. Netw.* 166:106988. doi: 10.1016/j.comnet.2019.106988

de Rocha, R. C. A., and Endler, M. (2006). "Middleware: context management in heterogeneous, evolving ubiquitous environments," in *IEEE Distributed Systems Online* (IEEE), 1–13. doi: 10.1109/MDSO.2006.28

del Castillo, A., Diebolt, P., Hazi, C., and Kreinin, M. L. I. W. (2020). *Industrial internet of things for monitoring services.* (MSc Group Project Report), School of Aerospace, Transport and Manufacturing, Cranfield University, 15–19.

Dey, A., Abowd, G., and Salber, D. (2001). A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. *Hum. Comput. Interact.* 16, 97–166. doi: 10.1207/S15327051HCI16234_02

Dhallenne, J., Jayaraman, P. P., and Zaslavsky, A. (2016). *RCOS: Real Time Context Sharing Across a Fleet of Smart Mobile Devices.* Cham: Springer International Publishing, 87–100. doi: 10.1007/978-3-319-46301-8_8

Dimitrova, V., Mehmood, M. O., Thakker, D., Sage-Vallier, B., Valdes, J., and Cohn, A. G. (2020). An ontological approach for pathology assessment and diagnosis of tunnels. *Eng. Appl. Artif. Intellig.* 90:103450. doi: 10.1016/j.engappai.2019.103450

Doukas, C., Capra, L., Antonelli, F., Jaupaj, E., Tamilin, A., and Carreras, I. (2015). "Providing generic support for IoT and M2M for mobile devices," in: *The 2015 IEEE RIVF International Conference on Computing Communication Technologies—Research, Innovation, and Vision for Future (RIVF)* (Can Tho: IEEE), 192–197. doi: 10.1109/RIVF.2015.7049898

Ebrahimipour, V., and Yacout, S. (2015). "Ontology-based schema to support maintenance knowledge representation with a case study of a pneumatic valve," in *IEEE Transactions on Systems, Man, and Cybernetics: Systems* (IEEE), 702–712. doi: 10.1109/TSMC.2014.2383361

Emmanouilidis, C., Koutsiamanis, R. A., and Tasidou, A. (2013). Mobile guides: taxonomy of architectures, context awareness, technologies and applications. *J. Netw. Comput. Appl.* 36, 103–125. doi: 10.1016/j.jnca.2012.04.007

Emmanouilidis, C., Pistofidis, P., Bertoncelj, L., Katsouros, V., Fournaris, A., Koulamas, C., et al. (2019). Enabling the human in the loop: linked data and knowledge in industrial cyber-physical systems. *Annu. Rev. Control.* 47, 249–265. doi: 10.1016/j.arcontrol.2019.03.004

Farghaly, K., Abanda, F. H., Vidalakis, C., and Wood, G. (2019). BIM-linked data integration for asset management. *Built Environ. Project Asset Manage.* 9, 489–502. doi: 10.1108/BEPAM-11-2018-0136

France-Mensah, J., and O'Brien, W. J. (2019). A shared ontology for integrated highway planning. *Adv. Eng. Inform.* 41:100929. doi: 10.1016/j.aei.2019.100929

Frank, A. G., Dalenogare, L. S., and Ayala, N. F. (2019). Industry 4.0 technologies: implementation patterns in manufacturing companies. *Int. J. Prod. Econ.* 210, 15–26. doi: 10.1016/j.ijpe.2019.01.004

Garlan, D., Siewiorek, D. P., Smailagic, A., and Steenkiste, P. (2002). 'Project aura: toward distraction-free pervasive computing. *IEEE Pervasive Comput.* 1, 22–31. doi: 10.1109/MPRV.2002.1012334

Gayathri, R., and Uma, V. (2018). Ontology based knowledge representation technique, domain modeling languages and planners for robotic path planning: a survey. *ICT Express* 4, 69–74. doi: 10.1016/j.icte.2018.04.008

Gil, D., Ferrández, A., Mora-Mora, H., and Peral, J. (2016). Internet of things: a review of surveys based on context aware intelligent services. *Sensors* 16:1069. doi: 10.3390/s16071069

Giurgiutiu, V., Adrian, C., and Goodman, P. (2001). *Review of Vibration-Based Helicopters Health and Usage Monitoring Methods.* Columbia: *University of South* Carolina, Department of Mechanical Engineering.

Gong, Y., and Janssen, M. (2013). An interoperable architecture and principles for implementing strategy and policy in operational processes. *Comput. Ind.* 64, 912–924. doi: 10.1016/j.compind.2013.06.018

Hassani, A., Medvedev, A., Haghighi, P. D., Ling, S., Indrawan-Santiago, M., Zaslavsky, A., et al. (2018). "Context-as-a-service platform: exchange and share context in an IoT ecosystem," in *2018 IEEE International Conference on Pervasive Computing and Communications Workshops, PerCom Workshops* (Athens: IEEE), 385–390. doi: 10.1109/PERCOMW.2018.8480240

Henricksen, K. (2003). *A framework for context-aware pervasive computing applications*. (Ph. D thesis). The School of Information Technology and Electrical Engineering, The University of Queensland, 13–20.

IEC 60812 (2018). *Failure Modes and Effects Analysis (FMEA and FMECA)* (Geneva: IEC).

ISO 10816-3 (2009). *Mechanical Vibration—Evaluation of Machine Vibration by Measurements on Non-Rotating Parts* (Geneva: ISO).

ISO 13306 (2017). *Maintenance—Maintenance Terminology* (Geneva: ISO).

ISO 13372 (2012). *Condition Monitoring and Diagnostics of Machines—Vocabulary* (Geneva: ISO).

ISO 13373-1 (2002). *Condition Monitoring and Diagnostics of Machines—Vibration Condition Monitoring* (Geneva: ISO).

ISO 13373-2 (2016). *Condition Monitoring and Diagnostics of Machines—Vibration Condition Monitoring* (Geneva: ISO).

ISO 17359 (2011). *Condition Monitoring and Diagnostics of Machines*: *General Guidelines* (Geneva: ISO).

ISO 2041 (2018). *Mechanical Vibration, Shock and Condition Monitoring—Vocabulary* (Geneva: ISO).

ISO 55000 (2014). *Asset Management—Overview*, *Principles and Terminology* (Geneva: ISO).

Jeschke S., Brecher C., Meisen T., zdemir D., and Eschert T. (2017). "Industrial internet of things and cyber manufacturing systems," in *Industrial Internet of Things. Springer Series in Wireless Technology,* eds S. Jeschke, C. Brecher, H. Song, and D. Rawat (Cham: Springer). doi: 10.1007/978-3-319-42559-7_1

Karray, M. H., Chebel-Morello, B., and Zerhouni, N. (2012). A formal ontology for industrial maintenance. *Appl. Ontol.* 7, 269–310. doi: 10.3233/AO-2012-0112

Keivanpour, S., and Ait Kadi, D. (2019). Internet of things enabled real-time sustainable end-of-life product recovery. *IFAC PapersOnLine* 52, 796–801. doi: 10.1016/j.ifacol.2019.11.213

Khan, M. A., Shahid, M. A., Ahmed, S. A., Khan, S. Z., Khan, K. A., Ali, S. A., et al. (2019). Gear misalignment diagnosis using statistical features of vibration and airborne sound spectrums. *Measurement* 145, 419–435. doi: 10.1016/j.measurement.2019.05.088

Koukias, A., Nadoveza, D., and Kiritsis, D. (2013). Semantic data model for operation and maintenance of the engineering asset. *IFIP Adv. Inform. Commun. Technol.* 398, 49–55. doi: 10.1007/978-3-642-40361-3_7

Lau, R., Li, C., and Liao, S. (2014). Social analytics: learning fuzzy product ontologies for aspect-oriented sentiment analysis. *Decis. Support Syst.* 65, 80–94. doi: 10.1016/j.dss.2014.05.005

Lawan, A., and Rakib, A. (2019). The semantic web rule language expressiveness extensions-a survey. *arXiv 1*1723.

Li, R., Mo, T., Yang, J., Jiang, S., Li, T., and Liu, Y. (2020). "Ontologies-based domain knowledge modeling and heterogeneous sensor data integration for bridge health monitoring systems," in *IEEE Transactions on Industrial Informatics* (IEEE), 3203. doi: 10.1109/TII.2020.2967561

Liu, C., Hua, J., and Julien, C. (2019). "SCENTS: collaborative sensing in proximity IoT networks," in *2019 IEEE International Conference on Pervasive Computing and Communications Workshops, PerCom Workshops* (Kyoto: IEEE), 189–195. doi: 10.1109/PERCOMW.2019.8730863

Liu, W., Li, X., and Huang, D. (2011). "A survey on context awareness," in *2011 International Conference on Computer Science and Service System (CSSS)* (Nanjing: IEEE), 144–147. doi: 10.1109/CSSS.2011.5972040

Lunardi, W. T., de Matos, E., Tiburski, R., Amaral, L. A., Marczak, S., and Hessel, F. (2015). "Context-based search engine for industrial IoT: discovery, search, selection, and usage of devices," in *2015 IEEE 20th Conference on Emerging Technologies* & *Factory Automation (ETFA),* (Luxembourg), 1–8. doi: 10.1109/ETFA.2015.7301477

Madhukalya, M., and Kumar, M. (2014). "ConCon: context-aware middleware for content sharing in dynamic participating environments," in *Proceedings - IEEE International Conference on Mobile Data Management,* (Brisbane, QLD: IEEE), 1, 156–161. doi: 10.1109/MDM.2014.25

Matsokis, A., and Kiritsis, D. (2012). "Ontology-based implementation of an advanced method for time treatment in asset lifecycle management," *in: Engineering Asset Management and Infrastructure Sustainability. eds J. Mathew, L. Ma, A. Tan, M. Weijnen, J. Lee (London, UK: Springer),* 647–662. doi: 10.1007/978-0-85729-493-7_50

Medina-Oliva, G., Voisin, A., Monnin, M., and Leger, J. B. (2014). Predictive diagnosis based on a fleet-wide ontology approach. *Knowledge-Based Syst.* 68, 40–57. doi: 10.1016/j.knosys.2013.12.020

Mineraud, J., Mazhelis, O., Su, X., and Tarkoma, S. (2016). A gap analysis of Internet-of-Things platforms. *Comput. Commun.* 89–90, 5–16. doi: 10.1016/j.comcom.2016.03.015

Noy, N. F., and McGuinness, D. L. (2001). A guide to creating your first ontology. *Biomed. Inform. Res.* 7–25. Available online at: https://protege.stanford.edu/publications/ontology_development/ontology101.pdf (accessed June 15, 2020).

Nuñez, D. L., and Borsato, M. (2018). OntoProg: an ontology-based model for implementing prognostics health management in mechanical machines. *Adv. Eng. Inform.* 38, 746–759. doi: 10.1016/j.aei.2018.10.006

Ong, S. K., and Zhu, J. (2013). A novel maintenance system for equipment serviceability improvement. *CIRP Ann. Manuf. Technol.* 62, 39–42. doi: 10.1016/j.cirp.2013.03.091

Papadopoulos, P., and Cipcigan, L. (2010). "Wind turbines condition monitoring: an ontology model," in *International Conference on Sustainable Power Generation and Supply* (Nanjing: IEEE), 1–4. doi: 10.1109/SUPERGEN.2009.5430854

Perera, C., Liu, C. H., Jayawardena, S., and Chen, M. (2015). A survey on internet of things from industrial market perspective. *IEEE Access* 2, 1660–1679. doi: 10.1109/ACCESS.2015.2389854

Perera, C., Zaslavsky, A., Christen, P., and Georgakopoulos, D. (2014). Context aware computing for the internet of things. *IEEE Commun. Surveys Tutorials* 16, 414–454. doi: 10.1109/SURV.2013.042313.00197

Perttunen, M., Riekki, J., and Lassila, O. (2009). Context representation and reasoning in pervasive computing: a review. *Int. J. Multimedia Ubiquit. Eng.* 4, 1–28.

Pradeep, P., and Krishnamoorthy, S. (2019). The MOM of context-aware systems: a survey. *Comput. Commun.* 137, 44–69. doi: 10.1016/j.comcom.2019.02.002

Ramachandran, G. S., and Krishnamachari, B. (2019). Towards a large scale IoT through partnership, incentive, and services: a vision, architecture, and future directions. *Open J. Int. Things 5,* 80–92.

Ren, G., Ding, R., and Li, H. (2019). Building an ontological knowledgebase for bridge maintenance. *Adv. Eng. Softw.* 130, 24–40. doi: 10.1016/j.advengsoft.2019.02.001

Rizou, S., Haussermann, K., Durr, F., Cipriani, N., and Rothermel,. K. (2010). "A system for distributed context reasoning," in *Sixth International Conference on Autonomic and Autonomous Systems (ICAS)* (Cancun: IEEE), 84–89. doi: 10.1109/ICAS.2010.21

Ruta, M., Scioscia, F., Ieva, S., Loseto, G., Gramegna, F., Pinto, A., et al. (2017). "Knowledge discovery and sharing in the IoT: the physical semantic web vision," in *Proceedings of the ACM Symposium on Applied Computing* (Marrakech), 492–498. doi: 10.1145/3019612.3019701

Sanislav, T., and Miclea, L. (2015). A dependability modeling approach for cyber-physical systems. *J. Comput. Sci. Control Syst.* 8, 23–28.

Schilit, B. N., and Theimer, M. M. (1994). "Disseminating active map information to mobile hosts." *IEEE Netw. 8,* 22–32. doi: 10.1109/65.313011

Sezer, O. B., Dogdu, E., and Ozbayoglu, A. M. (2018). Context-aware computing, learning, and big data in internet of things: a survey. *IEEE Internet Things J.* 5, 1–27. doi: 10.1109/JIOT.2017.2773600

Snidaro, L., García, J., and Llinas, J. (2015). Context-based information fusion: a survey and discussion. *Inform. Fusion* 25, 16–31. doi: 10.1016/j.inffus.2015.01.002

Statistica report. (2020). *Number of Internet of Things (IoT) Connected Devices Worldwide in 2018, 2025 and 2030.* Available online at: https://www.statista.com/statistics/802690/worldwide-connected-devices-by-access-technology/ (accessed June 22, 2020).

Tiburski, R. T., Amaral, L. A., Matos, E. D., Hessel, F. (2015). The importance of a standard security architecture for SOA-based IoT middleware. *IEEE Commun. Mag.* 53, 20–26. doi: 10.1109/MCOM.2015.7355580

Valverde-Rebaza, J. C., Roche, M., Poncelet, P., and de Lopes, A. (2018). The role of location and social strength for friendship prediction in location-based social networks. *Inform. Process. Manage.* 54, 475–489. doi: 10.1016/j.ipm.2018.02.004

van Bunningen, A. H., Feng, L., Apers, P. M. G. (2005). "Context for ubiquitous data management," *in Proceedings of the International Workshop on Ubiquitous Data Management, (Washington, DC),* 17–24. doi: 10.1109/UDM.2005.7

Wei, L., Du, H., Mahesar, Q. ain., Al Ammari, K., Magee, D. R., Clarke, B., et al. (2020). *A decision support system for urban infrastructure inter-asset management employing domain ontologies and qualitative uncertainty-based reasoning. Expert Syst. Appl. 158:113461.* doi: 10.1016/j.eswa.2020.113461

Wilson, D. H., Long, A. C., and Atkeson, C. (2005). "A context-aware recognition survey for data collection using ubiquitous sensors in the home," in *Conference on Human Factors in Computing Systems—Proceedings* (Portland, OR), 1865–1868. doi: 10.1145/1056808.1057042

Xu, L. D., He, W., and Li, S. (2014). Internet of things in industries: a survey. *IEEE Trans. Ind. Inform.* 10, 2233–2243. doi: 10.1109/TII.2014.2300753

Yamamoto, J., Nakagawa, H., Nakayama, K., Tahara, Y., and Ohsuga, A. (2009). "A context sharing message broker architecture to enhance interoperability in changeable environments," in *3rd International Conference on Mobile Ubiquitous Computing, Systems, Services, and Technologies* (Sliema: IEEE), 31–39. doi: 10.1109/UBICOMM.2009.48

Zhou, A., Yu, D., and Zhang, W. (2015). A research on intelligent fault diagnosis of wind turbines based on ontology and FMECA. *Adv. Eng. Inform.* 29, 115–125. doi: 10.1016/j.aei.2014.10.001

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

frontiers in Artificial Intelligence

REVIEW
published: 09 November 2020
doi: 10.3389/frai.2020.578613

Prognostics and Health Management of Industrial Assets: Current Progress and Road Ahead

Luca Biggio[1,2]* *and Iason Kastanis*[2]

[1]*Data Analytics Lab, Institute of Machine Learning, Department of Computer Science, ETHZ: Eidgenössische Technische Hochschule Zürich, Zurich, Switzerland,* [2]*Robotics and Automation, CSEM SA: Swiss Center for Electronics and Microtechnology S.A., Alpnach, Switzerland*

Prognostic and Health Management (PHM) systems are some of the main protagonists of the Industry 4.0 revolution. Efficiently detecting whether an industrial component has deviated from its normal operating condition or predicting when a fault will occur are the main challenges these systems aim at addressing. Efficient PHM methods promise to decrease the probability of extreme failure events, thus improving the safety level of industrial machines. Furthermore, they could potentially drastically reduce the often conspicuous costs associated with scheduled maintenance operations. The increasing availability of data and the stunning progress of Machine Learning (ML) and Deep Learning (DL) techniques over the last decade represent two strong motivating factors for the development of data-driven PHM systems. On the other hand, the black-box nature of DL models significantly hinders their level of interpretability, *de facto* limiting their application to real-world scenarios. In this work, we explore the intersection of Artificial Intelligence (AI) methods and PHM applications. We present a thorough review of existing works both in the contexts of fault diagnosis and fault prognosis, highlighting the benefits and the drawbacks introduced by the adoption of AI techniques. Our goal is to highlight potentially fruitful research directions along with characterizing the main challenges that need to be addressed in order to realize the promises of AI-based PHM systems.

Keywords: prognostic and health management, predictive maintenance, industry 4.0, artificial intelligence, machine learning, deep leaning

OPEN ACCESS

Edited by:
Dimitris Kiritsis,
École Polytechnique Fédérale de Lausanne, Switzerland

Reviewed by:
Saeed Tabar,
Ball State University, United States
Mehmet Ergun,
Istanbul Şehir University, Turkey

Correspondence:
Luca Biggio
lbiggio@student.ethz.ch

Specialty section:
This article was submitted to AI in Business, a section of the journal Frontiers in Artificial Intelligence

Received: *30 June 2020*
Accepted: *28 September 2020*
Published: *09 November 2020*

Citation:
Biggio L and Kastanis I (2020) Prognostics and Health Management of Industrial Assets: Current Progress and Road Ahead. Front. Artif. Intell. 3:578613. doi: 10.3389/frai.2020.578613

1 INTRODUCTION

Supporting the constant growth of modern industrial markets makes the optimization of operational efficiency and the minimization of superfluous costs essential. A substantial part of these costs often derives from the maintenance of industrial assets.

Recent studies[1] show that, for the average factory, inefficient maintenance policies are responsible for costs ranging from 5 to 20% of the plant's entire productive capacity. Furthermore, according to the International Society of Automation (ISA)[2], the overall burden of unplanned downtime on

[1]https://www2.deloitte.com/content/dam/Deloitte/us/Documents/process-and-operations/us-cons-predictive-maintenance.pdf

[2]https://www.isa.org/standards-publications/isa-publications/intech-magazine/2013/feb/automation-it-predictive-maintenance-embraces-analytics/

industrial manufacturers across all industry segments is estimated to touch the impressive figure of $647 billion per year.

If, on one hand, the above considerations highlight the fundamental impact of maintenance operations on manufacturers' balances, on the other hand a large number of companies are still not satisfied with their maintenance strategies. According to a recent trend study gathering interviews with more than 230 senior European business[3], roughly 93% of them deem their maintenance policy inefficient.

As discussed later, the current most popular approaches to maintenance are divided into two categories, namely reactive maintenance and scheduled maintenance. Roughly speaking, the first implements maintenance operations immediately after a system failure occurs, whereas the second is based on scheduling maintenance operations at regular time intervals. These strategies naturally introduce significant extra costs due to machine downtime, component replacement or unnecessary maintenance interventions.

On the other hand, Predictive Maintenance (PM) represents a completely different paradigm that holds the promise of overcoming the inefficiencies of the aforementioned methods. PM is one of the hallmarks of the so-called Industry 4.0 revolution, i.e., the process of modernization of the industrial world induced by the advent of the digitalization era. The goal of PM systems is to implement a smarter and more dynamical approach to maintenance leveraging recent advances in sensor engineering and data analysis. The health state of a machine is now constantly monitored by a network of sensors and future maintenance operations are based on the analysis of the resulting data. An increasing number of organizations, motivated by their need for reducing costs and by the potential of PM, are starting to invest significant amounts of resources on the modernization of their current maintenance strategies[1].

One natural question arising now is to what extent PM solutions can actually improve a company's efficiency in terms of reduction of downtime, cost savings and safety. A recent PWC study[4] investigates the actual potential of PM beyond the hype generated around it in the last few years. The results are quite impressive: 95% of the interviewed organizations claim that the adoption of PM strategies contributed to the improvement of several key performance indicators. Roughly 60% of the involved companies report average improvements of more than 9% of machines uptime, and further enhancements in terms of cost savings, health risks, assets lifetime.

As mentioned above, as a key player in the fourth industrial revolution, PM exploits some of the most recent advances introduced in the last few years in computer science and information engineering. Among them, ML is arguably one of the technologies that is experiencing the most impressive growth in terms of investments and interest of the private sector. This increasing attention in AI technologies is mainly due to the tremendous contributions they have brought in fields such as Computer Vision (CV), Natural Language Processing (NLP) and Speech Recognition in the last decade.

PM approaches are heavily based on ML techniques. The increasing availability of relatively cheap sensors has made much easier to collect large amounts of data, which are in turn the main ingredients ML systems necessitate.

However, AI-based technologies should not be considered as a "silver bullet" capable of immediately addressing all the issues affecting current maintenance strategies. ML and DL, in particular, are constantly evolving fields and, despite their significant achievements, a number of drawbacks still limit their wide application to real-world scenarios. It is, therefore, necessary to be cautious and try to understand the limitations of current AI approaches in the context of PM and drive further research toward the resolution or the alleviation of these shortcomings.

The goal of this manuscript is to provide an updated critical review of the main AI techniques currently used in the context of PM. Specifically, we focus on highlighting the benefits introduced by modern DL techniques along with the challenges that these systems are not yet able to solve. Furthermore, we present a number of relatively unexplored solutions to these open problems based on some of the most recent advances proposed in the AI community in the last few years.

This manuscript is structured as follows: **Section 2** briefly describes classic maintenance strategies and introduces the core ideas from Prognostic and Health Management (PHM). **Section 3** discusses the benefits of data-driven approaches and presents some of the most popular AI-based methods used in PHM. **Section 4** summarizes the main open challenges in PHM and presents some of their possible solutions. Finally, **Section 5** concludes the paper.

2 ELEMENTS OF PROGNOSTIC AND HEALTH MANAGEMENT

Prognostic and Health Management (PHM) is an engineering field whose goal is to provide users with a thorough analysis of the health condition of a machine and its components (Lee et al., 2014). To this extent, PHM employs tools from data science, statistics and physics in order to detect an eventual fault (anomaly detection) in the system, classify it according to its specific type (diagnostic) and forecast how long the machine will be able to work in presence of this fault (prognostic) (Kadry, 2012).

First, we present the most popular maintenance approaches, highlighting the advantages and disadvantages of these different methods in terms of costs and overall machine downtime. Then, we describe the entire PHM process by describing the role of its main sub-components in the context of the previously introduced maintenance approaches.

2.1 Different Approaches to Maintenance

The choice of an efficient maintenance strategy is crucial for reducing costs and minimizing the overall machine's downtime. The adoption of a particular maintenance strategy primarily

[3]https://www.ge.com/uk/sites/www.ge.com.uk/files/PAC-Predictive-Maintenance-GE-Digital-Full-report-2018.pdf

[4]https://www.pwc.be/en/documents/20180926-pdm40-beyond-the-hype-report.pdf

depends on the needs and the characteristics of the company's production line. Indeed, each maintenance policy introduces some benefits and disadvantages directly impacting costs in different modalities. In this review, we identify four distinct approaches to maintenance, namely: Reactive Maintenance (RM), Scheduled Maintenance (SM), Condition-Based Maintenance (CBM), and Predictive Maintenance (PM) (Fink, 2020).

2.1.1 Reactive Maintenance

RM consists of repairing or substituting a machine component only once it fails and it can no longer operate. The immediate advantage of this approach is that the amount of maintenance manpower and expenses related to keeping machines running are minimized (Swanson, 2001). Furthermore, since machines are active until they break, their utilization time is maximized. On the other hand, this approach is risky from many perspectives. First and foremost, it is potentially dangerous from the point of view of safety. Waiting for a machine to reach its maximum stress level can result in catastrophic failures. Moreover, this type of failures usually introduce larger costs and need a significant amount of time to be repaired. Therefore, by adopting this maintenance strategy, one might expect conspicuous costs arising both from reparations of severe failures and from relatively large unplanned machines downtimes.

2.1.2 Scheduled Maintenance

SM is based on maintenance interventions carried out at regular time intervals. The goal is to minimize the probability of failures and thus avoid costly unplanned downtimes by performing maintenance activities even when the machine is still operating under normal conditions. SM strongly relies on a meaningful schedule that has to be tailored to the specific properties of the equipment. In particular, experts have to provide a detailed evaluation of the failure behavior of the machines and of their components in order to maximize the level of accuracy on the prediction of the next failure time. This analysis typically results in the so-called "bathub" curves (Mobley, 2002), as shown in **Figure 1**.

The bathtub curve illustrated in **Figure 1** shows that a machine component presents a high risk of failure right after it is installed (because of installation errors or incompatibility issues with other components) and after its normal operation interval (because of natural degradation and wear out.). Between these two phases, the machine is supposed to work properly and its failure probability is low and constant.

The main advantage of SM is that it significantly reduce unplanned downtime. Furthermore, the reparation costs are generally less dramatic than those encountered in RM, since, now, machines are not allowed to operate until their breaking point. On the other hand, a SM approach presents the concrete risk of carrying out several relatively expensive maintenance interventions even when the equipment is still working properly. Sticking to a fixed degradation model of a certain machine might lead maintenance operators to miss anomalies caused by external factors or internal malfunctions that make the machine's degradation pattern deviate from its predicted trend.

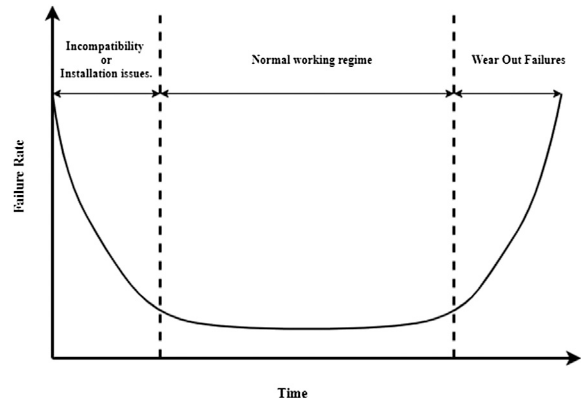


FIGURE 1 | The bathub curve shows that the most likely times for a machine to break are right after the installation and after its normal operating time.

2.1.3 Condition Based and Predictive Maintenance

CBM and PM differ from the types of maintenance strategies previously described in that they employ data-driven techniques to assist technicians to efficiently set times for maintenance activities. The goal of these methods is to provide a good compromise between maintenance frequency and its relative costs (Ran et al., 2019).

The difference between CBM and PM lies entirely in their different responses when a defective system condition is detected. In this case, a CBM approach would intervene on the system immediately after the detection time. This method could lead to the replacement or repair of a component of the equipment even if it could have continued its normal routine for a longer time without affecting other parts of the machine. Furthermore, intervening immediately after the fault has been detected might result in stopping the machines' working cycle at an inconvenient stage from the point of view of production efficiency.

To the contrary, PM tries to predict the useful lifetime of a component at a certain time step in order to indicate the point in the future where maintenance has to be performed. This last approach inevitably results in lower maintenance costs compared to CBM, since each component can be fully exploited without sacrificing safety and efficiency (Fink, 2020).

Figure 2 summarizes the maintenance strategies presented above by illustrating the costs resulting from their different approaches.

2.2 Prognostic and Health Management Process

As mentioned before, PHM makes use of information extracted from data to assess the health state of an industrial component and driving maintenance operations accordingly. **Figure 3** illustrates the main components constituting the typical PHM pipeline, from data acquisition to decision making.

The very first step of the PHM process consists of selecting a suitable set of sensors and devices, setting them up in the most appropriate location and deciding on an optimal sampling

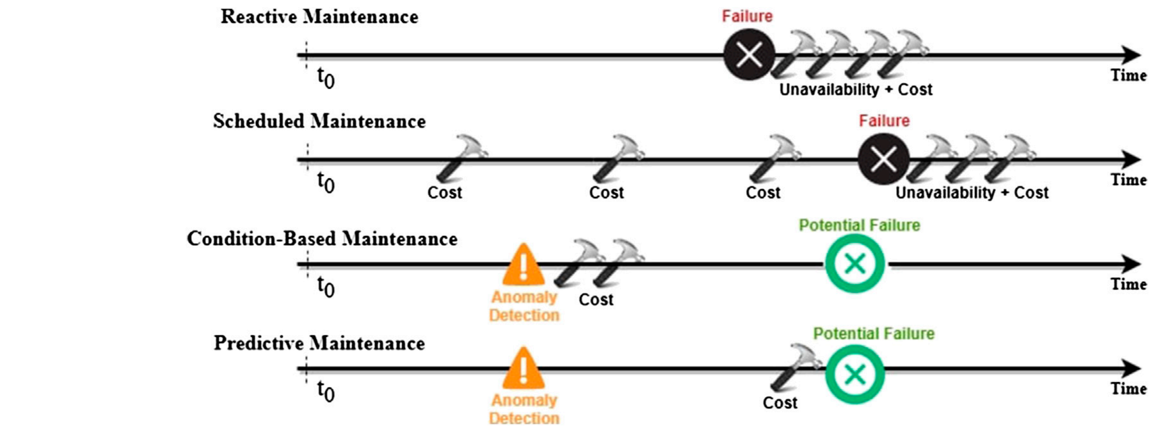


FIGURE 2 | Scheme of the behavior of the different maintenance approaches described above. Figure adapted from Fink (2020).

frequency for data collection. The communication system between sensors and databases must be implemented in order to allow for both real-time machine health monitoring and offline data handling. To this extent, a widely adopted solution by industries is the Open Platform Communication Unified Architecture (OPC UA), a popular communication protocol that allows information to be shared across sensors, industrial assets and the Cloud in a highly secure way (Bruckner et al., 2019).

Once the sensor array is in place, data can be acquired. These data are typically in forms that are not compatible with the input shape requested by AI algorithms. Therefore, a data pre-processing step must be implemented in order to clean the data, mitigate the effects induced by noise or simply reshape them so that their new format can be interpreted by data analysis techniques.

The resulting data are cleaner than the original ones but can still contain a substantial amount of redundant information. This motivates the application of feature extraction techniques to reduce the dimensionality of the data and retain only the most meaningful pieces of information. As we see in the next section, most modern AI techniques are designed to automatically extract informative features without any need for expert knowledge and manual feature engineering.

2.2.1 Condition-Based Maintenance

CBM consists of two main elements: anomaly detection and diagnosis [see **Figure 3** (left)]. Both these processes immediately

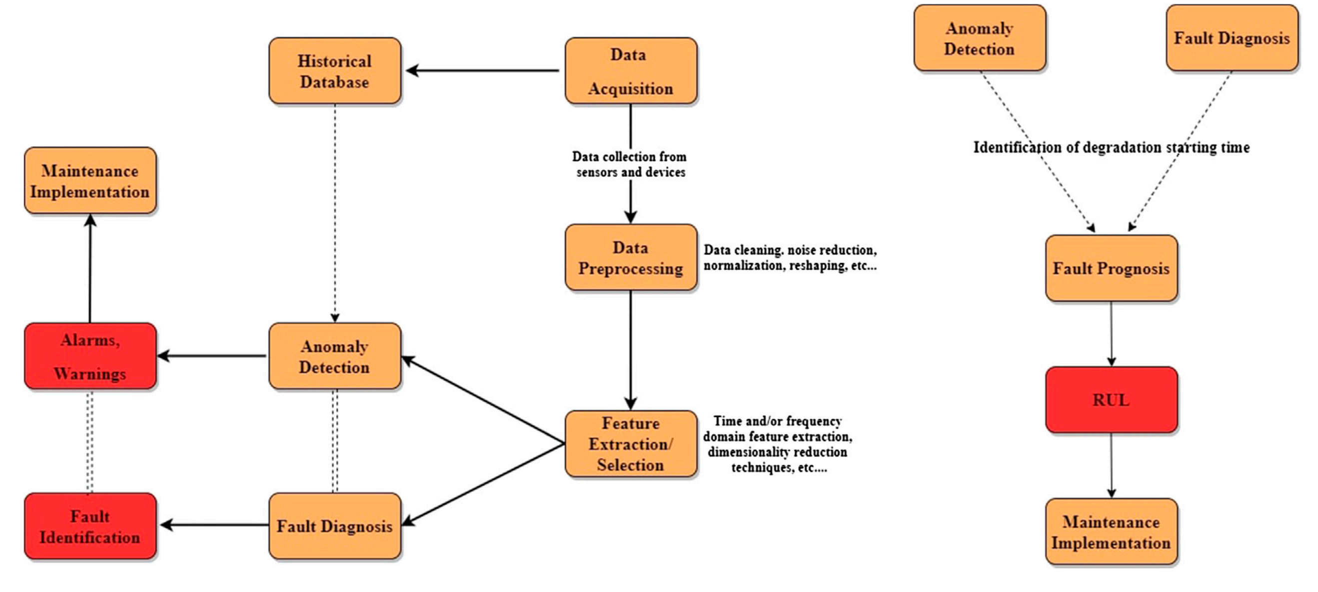


FIGURE 3 | Main steps of the typical PHM process. This can be divided into CBM **(left)** and PM **(right)**. RUL estimation is enhanced by information extracted at the CBM level, such as the time step where degradation starts to show its effects. Figure adapted from Khan and Yairi (2018).

follow the data extraction and data pre-processing pipelines described above and aim at supporting the decision making step with meaningful information about the state of the system. The information extracted by the anomaly detection and diagnosis modules can subsequently be exploited at the PM level in order to provide an even richer description of the machine's health state [see **Figure 3** (right)].

2.2.1.1 Anomaly Detection

Anomaly detection is responsible for automatically establishing whether the input data present any discrepancy compared to some internal model of the normal machine's behavior (Khan and Yairi, 2018). This internal representation can be learned by extracting and storing representative features from data gathered from healthy machines. It is important to note that, in general, healthy data, i.e., data gathered from machines working under normal working conditions, are much more abundant than faulty data. This is because, typically, a machine can incur in several different types of faults, each of which is, luckily, relatively rare. As a conclusive remark, we highlight that the detection of an anomaly does not necessarily imply that it corresponds to a fault. It might be, for instance, that it represents a new healthy feature that does not have any representatives into the historical data or has not been modeled by the anomaly detection algorithm's internal model.

2.2.1.2 Fault Diagnosis

Fault diagnosis moves one step forward with respect to anomaly detection since, besides detecting that an outlier is present, it also identifies the cause at the basis of that anomaly (Hess, 2002). Fault diagnosis models are based on historical data representing different faulty conditions. These data are used to characterize each type of fault and allow the models to classify new previously unseen data within a predefined set of fault cases.

2.2.2 Predictive Maintenance

The main difference between CBM and PM is that PM algorithms deal with the problem of predicting the Remaining Useful Life (RUL) of an industrial component before a complete failure occurs and the machine is no longer able to operate (Medjaher et al., 2012; Fink, 2020). Therefore, the key enablers of PM strategies are algorithms capable of efficiently forecasting the future state of a machine, i.e., provide prognostic information about its RUL.

2.2.2.1 Fault Prognosis

As mentioned before, fault prognosis is about providing an as accurate as possible prediction of the RUL of a certain machine component. The RUL estimation process starts from the identification of a time-step where a fault begins to show its effects. The final goal is to infer how long the machine can continue operating even in the presence of a degradation trend due to the previously detected fault.

Contrarily to diagnosis, time plays a crucial role in prognosis, since the objective is now to provide an estimate of the future time step when a certain event will occur (Lee et al., 2014). It is important to note that RUL predictions are strongly affected by various sources of noise. These can arise from noisy sensor readings, the inherent stochasticity of the RUL forecasting problem and the choice of an imperfect model for the machine degradation process.

3 ARTIFICIAL INTELLIGENCE-BASED PROGNOSTIC AND HEALTH MANAGEMENT

The attempt of devising artificial agents with the ability to emulate or even improve some aspects characterizing human intelligence is what makes AI an extremely exciting field of research both from a fundamental and a practical points of view. ML, as a branch of AI, studies the problem of designing machines capable of learning through experience and by extracting information from data (Mitchell, 1997). "Learning from experience" represents a distinctive human feature that enables us to actively interact with the world we live in. It allows us to build a progressively more accurate internal model of the surrounding environment by processing and interpreting the external signals our body is able to perceive.

Similarly to humans, intelligent systems can process the information perceived by an array of sensors about a given industrial component and provide a model of its operating condition and its health status. The increasing availability of data and the high level of computational power reached by modern hardware components make the application of AI techniques even more appealing.

ML has witnessed an increasing interest in the last few decades. A turning point has been set by the introduction of the first state-of-the-art DL technique almost 10 years ago by Krizhevsky et al. (2012) in the context of Image Recognition (IR). This event has triggered a new era in the field of data analysis characterized by a plethora of new applications of DL to a series of disparate engineering fields, ranging from NLP to CV.

The goal of this section is to give the reader an insight into the intersection of ML and PHM and the progress made by the scientific community hitherto. First, we present the main steps involved in the application of "traditional" ML techniques to PHM and we discuss how these can be utilized in the contexts of diagnosis and prognosis. Then, we present a number of popular DL techniques and we review some of their most interesting applications in this context.

3.1 "Classical" Machine Learning Methods

Before the explosion of DL almost one decade ago, the typical process followed by the majority of data-driven approaches to PHM is illustrated in **Figure 4**. The raw measurements provided by a battery of sensors can not be straightforwardly linked with the health state of the machine or its RUL. Indeed, they are often affected by a significant amount of noise that can be introduced by either external factors, such as a sudden temperature increase, or imperfect signal transmissions. Furthermore, often these data are represented by complex time-series or images, that are typically characterized by a highly redundant information content that tends to hide the relatively limited discriminative

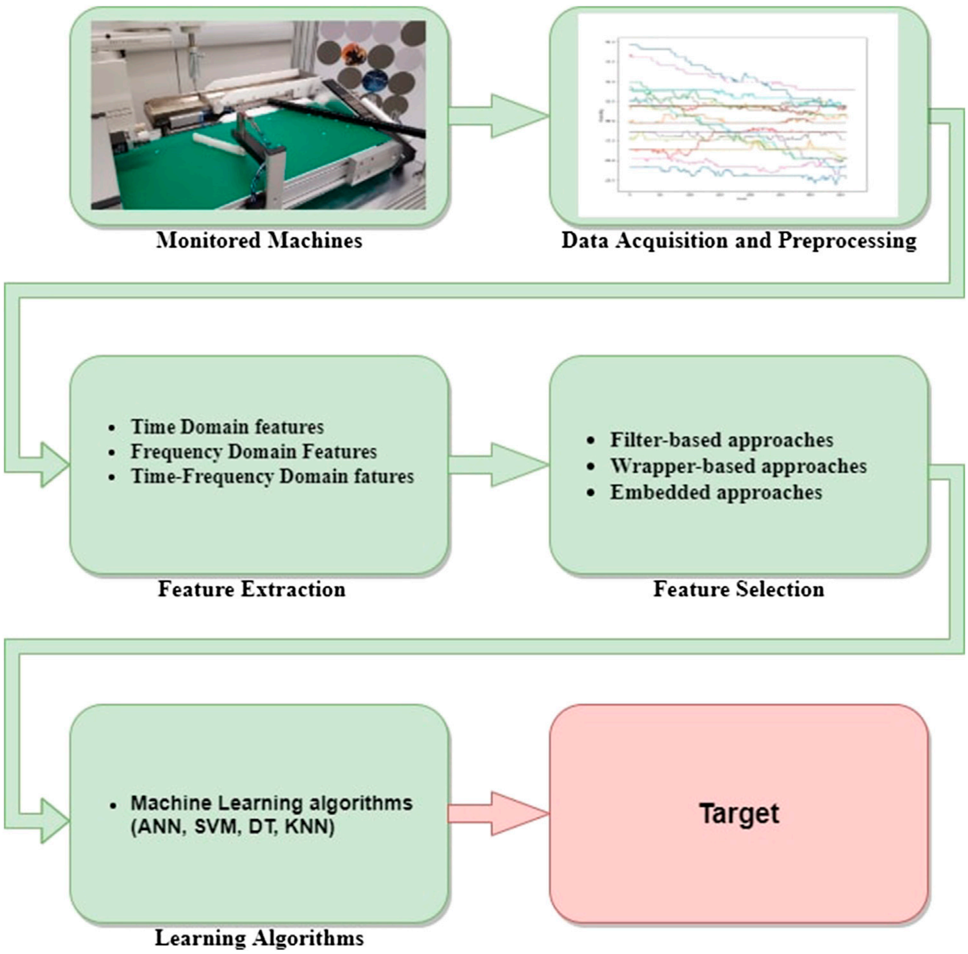


FIGURE 4 | Main steps characterizing the approaches based on traditional ML algorithms. Adapted from Zhao et al. (2016).

features of interest. For the above reasons, once data are acquired, a set of candidate features have to be extracted and then, only the most informative among them have to be properly selected. Once these steps are completed, the final set of extracted features can be used to train a ML algorithm to perform the desired diagnosis or prognosis task.

In the following, we briefly go through all the aforementioned steps, discussing some of the main techniques involved in each of them.

3.1.1 Feature Extraction and Feature Selection

3.1.1.1 Feature Extraction

According to Yu (2019), feature extraction can be defined as the task of transforming raw data into more informative features that serve the need of follow-up predictive models and that help in improving performances on unseen data.

A general recipe for the feature extraction task does not exist and a set of key context-dependent factors must be taken into account. Some of these are, for example, the specific type of task to be performed, the characteristics of the data, the application domain and the algorithmic and efficiency requirement (Guyon et al., 2006). For instance, traditional choices of features in the context of IR are those obtained by the SIFT (Lowe, 2004) and SURF (Bay et al., 2008) algorithms, whereas mel-cepstral coefficients (Davis and Mermelstein, 1980; Kopparapu and Laxminarayana, 2010) are typically chosen in speech recognition applications.

In the context of PHM, data recorded for the purpose of equipment maintenance come often in the form of time-series. Therefore, an opportune set of features must be chosen according to the properties of the signals under consideration, e.g., its physical nature (temperature, pressure, voltage, acceleration,. . .), its dynamics (cyclic, periodic, stationary, stochastic), its sampling frequency and its sample value discretization (continuous, discrete)[5]. Typical examples of features extracted from raw time-series data can be divided into three categories (Lei et al., 2020): time domain, frequency domain and time-frequency domain. The first includes statistical

[5]https://www.phmsociety.org/sites/phmsociety.org/files/Tutorial_PHM12_Wang.pdf

indicators such as mean, standard deviation, root mean square, skewness, kurtosis, crest factor, signal-to-noise ratio. Other standard time-domain feature extraction methods are traditional signal processing techniques such as auto and cross-correlation, convolution, fractal analysis (Yang et al., 2007) and correlation dimension (Logan and Mathew, 1996). Finally, model-based approaches such as autoregressive (AR, ARMA) or probability distribution models where features consist of the model parameters (Poyhonen et al., 2004) are also commonly used.

Features extracted from the frequency domain are typically obtained through spectral analysis of the signal of interest. Fast-Fourier-Transform is applied to raw data to extract the power spectrum and retrieve information about the characteristic frequencies of the signal. Finally, time-frequency domain feature extraction techniques include short-time Fourier transform, wavelet transform and empirical mode decomposition, among others. The goal of these methods is to capture how the frequency components of the signal vary as functions of time and are particularly useful for non-stationary time-series analysis.

3.1.1.2 Feature Selection

The goal of feature extraction is to obtain a first set of candidate features that are as informative as possible for the problem under consideration. Feature selection aims at reducing the dimension of the feature space by individuating a subset of features that are maximally relevant for a certain objective. According to the pioneering work of Guyon et al. (2006), feature selection methods can be divided into three categories: filters, wrappers and embedded methods. The first class of approaches consists of finding a subset of features that is optimal according to a specified objective measuring the information content of the proposed candidates. This objective is independent of the particular ML algorithm used to perform the PHM task and therefore the resulting features will be typically more general and potentially usable by different ML algorithms. Several feature selection techniques are based on the calculation of information-theoretic quantities such as the Pearson coefficient or the information gain. For instance, the Minimum-Redundancy-Maximum-Relevance (mRMR) technique is based on the idea that the optimal subset of features should be highly correlated with the target variable (which might be, for example, the classification label indicating a specific fault type) and mutually far away from each other.

Wrapper-based methods differs from their filter-based counterpart in the criteria they use for assessing the "goodness" of a specific set of features. Specifically, they directly employ the ML algorithm to get feedback, usually in form of accuracy or loss function, about the selected candidates. Wrappers are usually able to achieve better performances than filters since they are optimized with respect to a specific ML algorithm which is in turn tailored for a specific task. On the other hand, wrappers are biased toward the ML algorithm they are based on and therefore the resulting feature subset will not be generally adequate for alternative ML techniques.

The final class of feature selection methods is represented by the so-called embedded approaches. These techniques integrate the feature selection process directly into the ML algorithm in an end-to-end fashion. A popular example of embedded approach is the LASSO (Least Absolute Shrinkage and Selection Operator) (Tibshirani, 1996) which is a method for linear-regression that solves the following optimization problem:

$$\min_{w,b} \frac{1}{n} \sum_{i=1}^{n} \left(y_i - w^T x_i - b\right)^2 + \lambda w_1 \tag{1}$$

with

$$||w||_1 = \sum_{j=1}^{d} \left| w^{(j)} \right| \tag{2}$$

The $\mathscr{L}^1$ norm forces the learnt solution $\hat{w}$ to be sparse and therefore, only the least redundant features are selected. Other methods used for end-to-end feature selection are, for instance, the Akaike Information Criterion (AIC) (Sakamoto et al., 1986) and the Bayesian Information Criterion (BIC) (Neath and Cavanaugh, 2012) which are both based on finding features that are generalizable and not problem-specific.

As a conclusive remark, it is worth mentioning that, similarly to feature selection approaches, also dimensionality reduction methods aim at reducing the level of redundancy and maximizing the amount of informativeness present among the feature candidates. Techniques such as Principal Component Analysis (PCA) (Jolliffe, 1986) are used to project data onto a lower-dimensional linear subspace perpendicular to the feature removed. Other popular dimensionality reduction techniques are Linear Discriminants Analysis (LDA) (McLachlan, 2004), Exploratory Projection Pursuit (EPP) (Friedman, 1987), Independent Component Analysis (ICA) (Hyvärinen and Oja, 2000) and T-distributed Stochastic Neighbor Embedding (t-SNE) (Maaten and Hinton, 2008), among others.

3.1.2 Traditional Machine Learning Algorithms

As shown in **Figure 4**, once features are extracted and properly selected, they can be used as input for a ML algorithm responsible for performing the diagnosis or prognosis task we are interested in. In this section, we focus on "traditional" ML algorithms, i.e., popular AI methods widely employed before the advent of DL. These techniques can be divided into four main sub-categories, namely: (shallow) Artificial Neural Networks (ANNs), Support Vector Machines (SVMs), Decision Trees (DTs), and K-Nearest Neighbor (KNN).

3.1.2.1 Diagnosis

All the aforementioned classes of algorithms have been applied to fault diagnosis in several different contexts. In the following, we first briefly discuss the basic principles of these methods and then we list some of their most interesting applications.

3.1.2.1.1 Artificial Neural Networks. ANNs are popular ML algorithms whose design draws inspiration from the biological mechanism at the basis of neural connections in the human brain. They consist of elementary processing units, called neurons, connected to each other by means of dynamic weights of variable magnitudes, whose role is meant to emulate the behavior of synaptic connections in animals' brains. Different types ANNs topologies can be constructed by differently organizing the neurons and their relative connections. The choice of the specific ANN architecture crucially depends on the nature of the task to be performed, the data structure under consideration and the availability of computational resources.

Over the last two decades, ANNs have been used to detect and classify faults incurring in several diverse types of machines. For instance, they have been applied to fault diagnosis of rolling element bearings (Samanta and Al-Balushi, 2003), induction motors (Ayhan et al., 2006), gears (Samanta, 2004; Abu-Mahfouz, 2005), engines (Lu et al., 2001), turbine blades (Kuo, 1995; Ngui et al., 2017), electrical (Moosavi et al., 2016) and photovoltaic (Chine et al., 2016) devices, among others.

The choice of output layer directly reflects the kind of task we are interested in. For instance, for fault detection tasks, two neurons can be used to output the probability that the input corresponds to a healthy instance or a faulty one. On the other hand, if we are interested in fault diagnosis, the number of output neurons is equal to the number of faults affecting the machine under consideration. A typical example of ANNs application to fault detection is provided by Samanta and Al-Balushi (2003). In this work, five time-domain features (RMS, skewness, variance, kurtosis, and normalized sixth central moment) are extracted from raw vibration signals. These features are then used as inputs to a shallow ANN consisting of two hidden layers with 16 and 10 neurons respectively and one output layer with two neurons (indicating if the input corresponds to normal or failed bearing).

3.1.2.1.2 Support Vector Machines. Given a dataset $\{x_i, y_i\}_{i=1}^{N}$, where $x_i \in \mathbb{R}^{d}$ and $y = \pm 1$, SVMs aim at separating the two classes of data by finding the optimal hyperplane with the maximum margin between them. The margin is the distance between the nearest training data points of any class. In most real-world problem, data are not linearly separable. In these cases, the so-called kernel trick (Hofmann et al., 2008) can be used to tackle nonlinear classification tasks by implicitly mapping the data into a high-dimensional feature space.

Standard SVMs, along with a number of improved variants, have been extensively applied to fault diagnosis. For example, they have been used for assessing the health state of rolling element bearings (Yang et al., 2005; Abbasion et al., 2007; Gryllias and Antoniadis, 2012; Fernández-Francos et al., 2013; Islam et al., 2017; Islam and Kim, 2019b), induction motors (Widodo and Yang, 2007), gearboxes (Liu et al., 2013), engines (Li et al., 2012), wind turbines (Santos et al., 2015) and air conditioning systems (Sun et al., 2016a).

In order to perform fault diagnosis tasks, SVMs are typically employed alongside One-Against-One (OAO) (Yang et al., 2005; Islam et al., 2017) or One-Against-All (OAA) (Abbasion et al., 2007; Gryllias and Antoniadis, 2012) strategies. Furthermore, SVMs can also be applied to anomaly detection. For example, Liu et al. (2013) train a one-class SVM only on healthy data to detect anomalies in bearings vibrational data.

Generally, SVMs are particularly well suited for problems characterized by high-dimensional features. On the other hand, the computation of the $N \times N$ kernel matrix can be highly expensive when the number of data instances is relatively large.

3.1.2.1.3 Decision Trees. Decision trees (DTs) represent a class of non-parametric supervised ML algorithms commonly used for regression and classification. DTs are trained to infer a mapping between data features and the corresponding output values by learning a set of relatively simple and interpretable decision rules. As the name suggests, these classification rules correspond to paths linking the root node to the leaf nodes. Indeed. each internal node can be seen as a condition on a particular attribute. The different outcomes of this test are represented by the branches generated from that node. The C4.5 algorithm (Quinlan, 2014) is one of the most popular approaches to learn a DT.

DTs have been widely employed in the context of fault diagnosis over the last two decades. For example, they have been applied to process data gathered from rolling element bearing systems (Sugumaran and Ramachandran, 2007; Sugumaran, 2012), gearboxes (Saravanan and Ramachandran, 2009; Praveenkumar et al., 2018), wind turbines (Abdallah et al., 2018), centrifugal pumps (Sakthivel et al., 2010), and photovoltaic systems (Benkercha and Moulahoum, 2018).

Multiple DTs can be employed jointly to form a random forest (RF), an ensemble learning algorithm capable of overcoming some shortcomings of single decision trees, such as limited generalization and overfitting. RFs have been successfully applied to fault diagnosis of induction motors (Yang et al., 2008), rolling bearings (Wang et al., 2017), and aircraft engines (Yan, 2006) among others.

The main advantages provided by DTs stand in their high level of interpretability, resulting from the easily decipherable decision rules they implement. Moreover, they often achieve reasonably high accuracies in most of the classification problems they are applied to. On the other hand, these methods are often prone to overfitting and therefore tend to provide poor generalization performances.

3.1.2.1.4 K-Nearest Neighbor. KNN is non-parametric algorithm widely used for classification tasks. Given a set of input-output pairs $\{x_i, y_i\}_{i=1}^{N}$ and a test datum $\hat{x}$, the KNN algorithm searches the k closest training inputs to $\hat{x}$ in the feature space and label the test datum with the label having more representatives among the k selected training data. Closeness can be measured by an arbitrary similarity measure, such as the Euclidean distance. Due to its simplicity and its high level in interpretability, KNN-based approaches have found many applications in fault diagnosis. For example, the literature includes example of applications in the context of rolling element bearings (Mechefske and Mathew, 1992; Moosavian et al., 2013; Tian et al., 2016) and gears (Lei and Zuo, 2009; Gharavian et al., 2013).

Enhanced versions of the basic KNN algorithms have been gradually introduced to boost its classification performances and to overcome some of its limitations, such as the computational load it requires to process large-sized datasets. For instance, Appana et al. (2017) introduce a new type of metric which augments the information provided by the distance between sample pairs with their relative densities. Also, Lei et al. (2009) apply a combination of weighted KNN (WKNN) classifiers to fault diagnosis of rolling bearings in order to cope with the problem of data instances belonging to different classes overlapping in the feature space. Finally, in Dong et al. (2017) and (Wang and Ma, 2014), KNN was optimized with the particle swarm algorithm (Kennedy and Eberhart, 1997) to alleviate the storage requirements of the former.

Overall, KNN and its enhanced versions can be considered as relatively effective algorithms for fault diagnosis, especially because of their simplicity and interpretability. Their main limitations stand in the high computational cost and their considerable sensitivity to noise.

3.1.2.2 Prognosis

Generally, prognosis is a more challenging problem than diagnosis and therefore effective methods in this context are less simple to find. Below, we list some of the most interesting applications of ANNs, SVMs, and DTs to fault prognosis. KNNs are not as widespread as in fault diagnosis and their application is not common in RUL estimation.

3.1.2.2.1 Artificial Neural Networks. Two of the first attempts of applying ANNs to fault prognosis problems are introduced in Shao and Nezu (2000) and Gebraeel et al. (2004). Both approaches are proposed in the context of bearings RUL prediction. In Shao and Nezu (2000), a three-layer neural network is used to forecast the value of the bearing health indicator. In Gebraeel et al. (2004) several fully-connected models are trained on either individual or on clusters of similar bearing features. Both methods use manually extracted statistical features as input of the corresponding ANNs. More recent approaches include, for example, Elforjani and Shanbr (2018) and Teng et al. (2016). The first work proposes a comparative study of the performance of SVM, Gaussian Processes (Rasmussen, 2003) and ANNs for RUL estimation from features extracted from acoustic emission signals. The study reveals that the proposed ANN is the best performing model for the RUL prediction task under consideration. In Teng et al. (2016), ANNs are used to provide short-term tendency prediction of a wind turbine gearbox degradation process. The approach is validated by a series of experiments on bearing degradation trajectories datasets, showing good RUL prediction performances.

3.1.2.2.2 Support Vector Machines. SVM-based methods have been extensively applied to fault prognosis tasks. Huang et al. (2015) provide an extensive review of the most relevant techniques employing SVM-related approaches in the context of RUL prediction. Application examples include RUL estimation of bearings (Sun et al., 2011; Chen et al., 2013; Sui et al., 2019), lithium-ion batteries (Khelif et al., 2017; Wei et al., 2018; Zhao H. et al., 2018; Zhao Q. et al., 2018) and aircraft engines (Ordóñez et al., 2019). For instance, in Wei et al. (2018) Support Vector Regression (SVR) is used to provide a state-of-health state-space model capable of simulating the battery aging mechanism. Comparison of the performances provided by an ANN-based model of the same type shows the superiority of the proposed approach over its neural network-based counterpart. In the context of bearings fault prognosis, Sun et al. (2011) introduce a multivariate SVM for life prognostics of multiple features that are known to be tightly correlated with the bearings' RUL. The proposed method shows good prediction performance and leverages the ability of SVM of dealing with high-dimensional small-sized datasets.

3.1.2.2.3 Decision Trees. DTs and RFs have also been applied to fault prognosis, in particular in the contexts of RUL estimation of bearings (Satishkumar and Sugumaran, 2015; Patil et al., 2018; Tayade et al., 2019), lithium-ion batteries (Zheng H. et al., 2019; Zheng Z. et al., 2019) and turbofan engines (Mathew et al., 2017). In Patil et al. (2018), the authors train a RF to perform RUL regression by using time-domain features extracted from the bearings vibration signals. The model is evaluated on the dataset provided by IEEE PHM Challenge 2012 (Ali et al., 2015), showing improved results than previous benchmarks. One further example is provided by Satishkumar and Sugumaran (2015), who cast the RUL estimation problem into a classification framework. In particular, statistical features in the time domain are extracted from five different temporal intervals from normal condition to bearing damage. A DT is then used to classify new data into one of these intervals, resulting in about 96% accuracy.

3.1.3 Discussion

3.1.3.1 Dependency on Feature Extraction

Traditional ML algorithms have been widely applied both to fault diagnosis and fault prognosis tasks. They present the relevant advantage of combining rather good performances and a relatively high degree of interpretability. On the other hand, most of them rely on good quality features that have to be carefully extracted and selected by human experts. This dependency on the feature extraction step limits the potential of traditional ML methods and imposes a strong inductive bias in the learning process. As we discuss in the next section, "deep" algorithms can extract information directly from raw data and can often improve the generalization performances of traditional ML approaches.

3.1.3.2 Model Selection

It is important to observe that it is not possible to identify a specific algorithm, among those discussed above, that clearly outperforms the others in all possible settings. Selecting a specific technique highly depends on the requirements and characteristics of the PHM problem at hand. For example, a black-box ANN approach might be more suitable when one is mainly interested in performances and less in interpretability, SVMs can be useful in the low-data regime and DTs can be a sensible choice if interpretability is prioritized. Ultimately, the final algorithm is often chosen by calculating a set of performance metrics for each candidate technique and selecting the method providing the highest scores. Some standard example of these measures are accuracy, precision, Recall, F1 Score, Cohen Kappa

(CK), and Area Under Curve (AUC). A description of these metrics can be found, for instance, in Bashar et al. (2020).

3.1.3.3 Overfitting

The long-standing problem of overfitting (or over-training) is a well-known pathology affecting data-driven approaches. In essence, it stems from the imbalance between model capacity and data availability. If on one hand, the adoption of ML techniques can be significantly beneficial in PHM, on the other hand, it also requires to think about effective solutions to contrast overfitting in order to fully exploit the advantages of data-driven approaches. In the context of PHM applications, a key requirement for the deployment of a given ML algorithm stands indeed in the robustness of its performances when data different from the training ones kick in. Although algorithm-specific techniques exist to tackle overfitting, held-out-cross validation (Hastie et al., 2001) is probably the most popular one and can be used independently on the particular ML algorithm (see, for instance, Gebraeel et al., 2004), for ANNs (Islam et al., 2017), for SVMs (Abdallah et al., 2018), for decision trees and (Tian et al., 2016) for KNN).

As regards DTs, overfitting is typically tackled by pruning the tree in order to prevent it to merely memorize the training set and improve performances on unseen data (Praveenkumar et al., 2018). Random forests have also been used for the same purpose (Yang et al., 2008). They consist of ensembles of DTs and one of their main benefits is to mitigate the overfitting tendency of standard DTs.

A widely used strategy to contrast over-training in SVMs is to introduce a set of so-called slack variables in order to allow some data instances to lie on the wrong side of the margin (Hastie et al., 2001). The extent to which this class overlapping effect is permitted is regulated by a regularization constant C. Furthermore, the smoothness of the margin can be adjusted by appropriately tuning the hyperparameters of the kernel. Sun et al. (2016a), for instance, use cross validation to find optimal values of the constant C and of the gaussian kernel width parameter.

In ANNs, the effects of overfitting get increasingly more pronounced as the number of hidden layers increases (Samanta, 2004). Two typical strategies to alleviate its impact are early stopping and regularization. The first consists in stopping the training phase once the first signs of over-training kick in. The second introduces a penalizing term in the loss function (typically in the form of $\mathbb{L}_2$ or $\mathbb{L}_1$ norms on the network weights) to keep the values of the weights as small as possible. In Ayhan et al. (2006) for instance, the authors use early-stopping by arresting the training phase once the validation error keeps increasing for a specific number of epochs.

Finally, the KNN algorithm yields different performances depending on the value of k. Small values of k result in very sharp boundaries and might lead to overfitting. On the other hand, large ks are more robust to noise but might result in poor classification performances. This hyperparameter is then typically chosen via cross-validation by selecting the best performing value among a set of candidates. In Gharavian et al. (2013), for instance, K is varied from 1 to the number of the training samples.

3.2 The Deep Learning Revolution

Most of the methods we have discussed so far are characterized by relatively "shallow" architectures. This aspect results in two main consequences: first, their representational power can be fairly limited and second, their input often consists of high-level features manually extracted from raw data by human experts.

DL is a quite recent class of ML methods that provide a new set of tools that are able to cope with the aforementioned shortcomings of traditional approaches. Essentially, DL techniques arise as an extension of classical ANNs. DL models, in their simplest form, can be seen as standard ANNs with the addition of multiple hidden layers between the network's input and output. An increasingly large corpus of empirical results has shown that these models are characterized by a superior representational power compared to shallow architectures. Once deep networks are trained, their inputs pass through a nested series of consecutive computations, resulting in the extraction of a set of complex features that are highly informative for the task on interest. This characteristic is one of the hallmarks of DL and can be seen as one of the key factors of its success.

In light of its improved representational power, its ability to automatically extract complex features, its dramatic achievements across different engineering fields and its multiple dedicated freely available software libraries (Jia et al., 2014; Abadi et al., 2016; Theano Development Team, 2016; Paszke et al., 2019), DL has the potential to provide effective solutions also in the context of PHM applications. Big data handling, automated end-to-end feature extraction from different data structures (e.g., images, time-series) and improved generalization are some of the targets on which DL models can make a difference compared to traditional ML approaches.

In the following, we introduce some of the most popular DL techniques used in PHM. Specifically, we focus on Autoencoder (AE) architectures, Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs) and some of their variants and combinations. For each model, we list some interesting applications both in the context of fault diagnosis and prognosis.

3.2.1 Methods and Techniques

3.2.1.1 Autoencoders

AEs, in their simplest form, consist of feed-forward neural networks that are trained to output a reconstructed version of their input. They are composed of two sub-networks, namely an encoder and a decoder. The encoder, h, implements a mapping from the input space to a typically lower-dimensional space. More concretely, we have:

$$\mathrm{h} = \psi(\mathrm{W}_1\mathrm{x} + \mathrm{b}_1) \tag{3}$$

where $\mathrm{x} \in \mathbb{R}^{\mathrm{d}}$ is the input vector, ψ is the activation function and $\mathrm{W}_1 \in \mathbb{R}^{\mathrm{q}\times\mathrm{d}}$ and $\mathrm{b}_1 \in \mathbb{R}^{\mathrm{q}}$ are the parameters of the encoder. The decoder implements a mapping from the embedding to the input space in order to reconstruct the original input vector. In formulas:

$$\tilde{x} = \psi(W_2 h + b_2) \tag{4}$$

where $\tilde{x} \in \mathbb{R}^d$ is the reconstructed input vector and $W_2 \in \mathbb{R}^{d\times q}$ and $b_2 \in \mathbb{R}^d$ are the parameters of the decoder. Given a dataset of N data instances $\{x_i\}_{i=1}^N$, the accuracy of the model can be measured with, for example, the Root-Mean-Squared-Error (RMSE), which evaluates the reconstruction error made by the autoencoder:

$$\mathrm{RMSE}(\theta) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(x_i - \tilde{x}_i(\theta)\right)^2} \tag{5}$$

In the equation above, the symbol θ has been used to indicate the parameters of the network, i.e., W_1, W_2, b_1, b_2. The value of the parameters is found by minimizing the RMSE w. r.t the parameter θ of the model. **Figure 5** shows an illustration of the typical AE architecture.

Note that the model assumes a so-called bottle-neck shape, characterized by an embedding space with a lower dimension than the input space. By setting $q < d$, we can force the algorithm to find a more expressive representation of the input by getting rid of redundant pieces of information and keep only the most relevant ones for the reconstruction purpose. It is important to point out that here we have limited our description to a one-hidden-layer architecture for the sake of simplicity. However, deep models can be simply obtained by consecutively stacking multiple hidden layers. following the bottle-neck architecture.

There exists several more powerful extensions of the basic AE discussed before. Some examples include Sparse AEs (SAEs) (Ng et al., 2011), denoizing AEs (DAEs) (Vincent et al., 2008) and variational AEs (VAEs) (Kingma and Welling, 2013). Sparse AEs regularize the standard AE loss function with an additional term that forces the model to learn sparse features. This regularization term can be, for instance, the $\mathbb{L}^1$ norm of the activations:

$$\mathrm{Loss}(\theta) = \mathrm{RMSE}(\theta) + \lambda\sum_i \left|h_i\right|, \tag{6}$$

where h_i is the ith component of the embedding h. Alternatively, one can consider the KL divergence between the average ith activation and a small sparsity parameter α, yielding the following loss:

$$\mathrm{Loss}(\theta) = \mathrm{RMSE}(\theta) + \lambda\sum_i KL\left(\alpha \| \rho_i\right), \tag{7}$$

where $\rho_i = \sum_j^m h_i(x_j)$ and m is the number of training examples.

DAEs take as input corrupted version of the data and aim to output a reconstructed version of the original uncorrupted data. The assumption is that the algorithm is forced to select only the most informative part of the input distribution in order to recover the uncorrupted data instance.

VAEs differ from the previous AE techniques since they belong to the class of generative models. They aim at learning a parametric latent variable model through the maximization of a lower bound of the marginal log-likelihood of the training data.

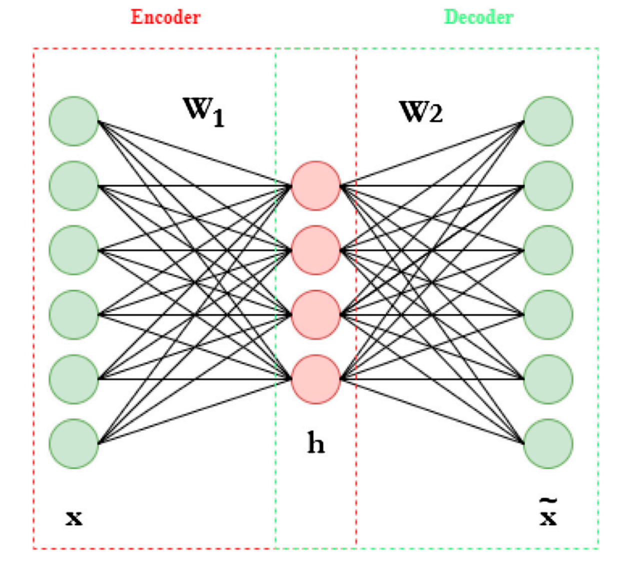


FIGURE 5 | Typical Autoencoder architecture.

The goal of these approaches is to provide a way to learn a so-called disentangled representation of the latent space, i.e., a representation where the most relevant independent factors of variations in the data are decoupled amd clearly separated. To conclude this part it is worth mentioning that it is possible to design autoencoders where the encoder and the decoder are not limited to simple feed-forward neural networks but can also assume the form of CNNs and RNNs. We discuss these methods later within the section.

3.2.1.2 Convolutional Neural Networks

CNNs are some of the most successful and widely applied DL models. They reached the peak of their popularity thanks to their state-of-the-art performances in CV tasks, such as IR, pose estimation and object tracking. They have also been successfully applied in the contexts of NLP, Reinforcement Learning and time-series modeling. Their design draws inspiration from the organization of animal visual cortex (Hubel and Wiesel, 1968). Indeed, it turns out that single cortical neurons fire in response of stimuli received from relatively narrow regions of the visual field called receptive fields. Furthermore, neurons that are close to each other are often associated with similar and partially overlapping receptive fields, allowing them to map the whole visual field. These properties are useful to recognize specific features in natural images independently of their location.

CNNs implement these concepts by modifying the way computations are usually performed in standard feed-forward neural networks. In particular, CNNs convolve the input image with filters composed of learnable parameters. These parameters are trained to automatically extract features from the image in order to perform the task specified by a final loss function.

The standard CNN model shown in **Figure 6** is composed of a set of elementary consecutive blocks. First, the input layer defines

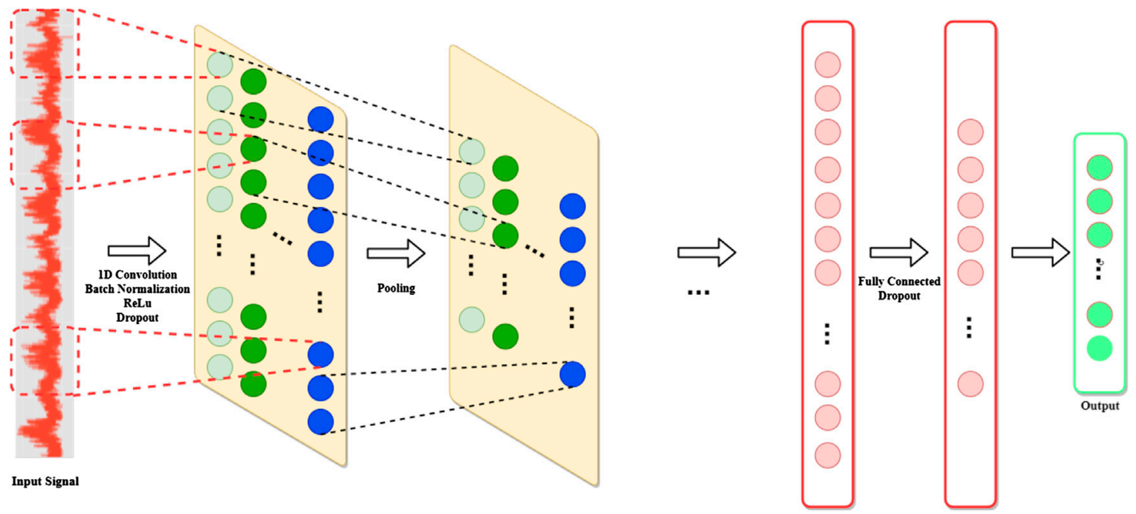


FIGURE 6 | Typical 1D-CNN architecture. Adapted from Jiao et al. (2020).

the data structure. A convolutional layer follows the input layer and performs the convolution operation over the input data. The size of the filters depend on the input structure. Two-dimensional filters are used for grid-like inputs, whereas, one-dimensional filters are used for time-series. Each filter has a user-specified size, which defines its receptive field. Batch normalization (Ioffe and Szegedy, 2015) is often applied right after the convolutional module in order to reduce the so-called covariate shift phenomenon and introduce a regularization effect. Then, a point-wize nonlinear activation function (e.g., ReLU) is applied.

The convolutional layer is then followed by a so-called pooling layer, whose role is to reduce the number of parameters by sub-sampling the filtered signals. One common strategy to perform this operation is called max-pooling and consists of extracting only the maximum value of a fixed-sized batch of consecutive inputs.

Several instances of convolutional and pooling layers are typically alternated through the network. The final filtered signals are then flattened and fed into a sequence of fully-connected layers that map them into the output layer. The dropout (Srivastava et al., 2014) technique can be used both between the fully connected and the convolutional layers in order to contrast overfitting.

3.2.1.3 Recurrent Neural Networks

RNNs form another class of DL methods that has achieved impressive results in a wide variety of ML fields. In particular, RNNs are particularly effective in processing data characterized by a sequential structure. These types of data are widespread in fields such as NLP, Speech Recognition, Machine Translation, Sentiment Analysis to name a few, where recurrent architectures have been employed successfully. Given their particular suitability in analyzing sequential data, it is not surprising that RNN models have been widely applied in the context of PHM applications. We review some of these applications later in this section.

The architecture of the simplest possible recurrent model is shown in **Figure 7**.

Given a sequential input vector $x = [x_1, \ldots, x_t, \ldots, x_T]$, where $x_t \in \mathbb{R}^d$ at each time-step the RNN shown above performs the following operations:

$$\begin{aligned} h_t &= \psi_1 (W_1 x_t + W_2 h_{t-1} + b_1) \\ o_t &= \psi_2 (W_3 h_t + b_2) \end{aligned} \tag{8}$$

where, W_1, W_2, W_3, b_1, b_2 are the parameters of the model, ψ_1 and ψ_2 are activation functions, h_t is the so-called hidden state at time t and o_t is the output at time t. Predictions are performed at each time step by mapping the current hidden state to the output. o_t, through a nonlinear activation. The hidden state is constantly updated at each iteration by combining the previous hidden state and the current input. This allows us to store past information and propagate it over time through the network. The basic

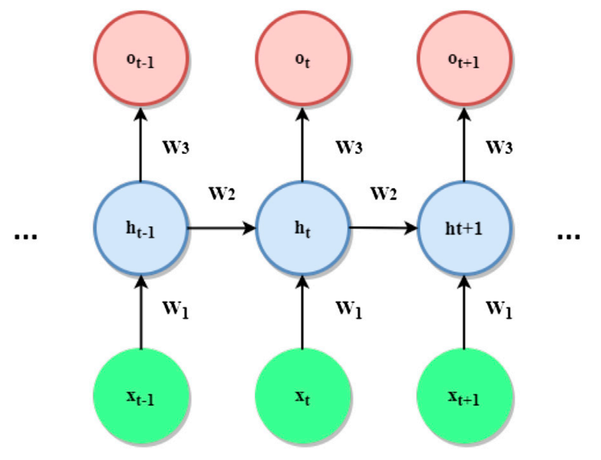


FIGURE 7 | Most elementary RNN architecture.

architecture described above, however, suffers from the so-called vanishing gradient problem. This phenomenon is caused by the structure of simple RNNs which typically perform the composition of the same function sequentially at each time step. As shown by Bengio et al. (1994), this results in increasingly small magnitudes associated with the gradients of long term interactions. To cope with this problem, a number of refinements have been introduced to the elementary architecture discussed before. The most popular ones are arguably the Long-Short-Term-Memory (LSTM) (Hochreiter and Schmidhuber, 1997), Bidirectional RNNs (Bi-RNN) (Schuster and Paliwal, 1997) and Gated-Recurrent Units (GRUs) (Cho et al., 2014). These techniques have been largely applied, over the last few years, to PHM, both for diagnosis and prognosis tasks. Current state-of-the-art methods in NLP complement the aforementioned recurrent architectures with the so-called attention mechanism (Devlin et al., 2018), which has resulted in significant performance improvements. Despite its success in NLP and related fields, attention-based networks do not find many applications in PHM, indicating a probably fruitful research direction.

3.2.2 Diagnosis

3.2.2.1 Autoencoder

AEs provide a first example of how DL methods can overcome some of the limitations of classical approaches. Indeed, typically AEs are used to automatically extract complex and meaningful features from raw data or to obtain more informative representations of a set of already extracted features. AEs have been applied to data gathered from several machines and industrial components, such as rolling element bearings (Jia et al., 2016; Liu et al., 2016; Lu et al., 2016; Jia et al., 2018), gearboxes (Jia et al., 2018), electrical generators (Michau et al., 2017; Michau et al., 2019), wind turbines (Yang et al., 2016), chemical industrial plants (Lv et al., 2017), induction motors (Sun et al., 2016b), air compressors (Thirukovalluru et al., 2016), hydraulic pumps (Zhu et al., 2015), transformers (Wang et al., 2016), spacecrafts (Li and Wang, 2015) and gas turbine combustors (Yan and Yu, 2019).

As mentioned before, AEs are often used in combination with other classifiers, such as simple softmax classifiers (Liu et al., 2016), feed-forward neural networks (Sun et al., 2016b), RFs (Thirukovalluru et al., 2016) and SVMs (Sun et al., 2016b; Lv et al., 2017). In Sun et al. (2016b), feed-forward NNs trained on top of the features learned by the AE model provide excellent classification results in terms of fault diagnosis accuracy. An SVM trained on the same features performs only slightly worse. Liu et al. (2016) propose a combination of stacked SAEs and a softmax classifier for element bearings fault diagnosis. Short-time-Fourier transformed raw inputs undergo several nonlinear transformations implemented by the sparse AEs. The resulting features are fed into a softmax classifier which outputs the classification results.

Lu et al. (2016) compare the features extracted by stacked DAEs with some manually extracted features. The comparison is based on the fault classification accuracies provided by an SVM and a RF model trained on top of the two classes of features. The results show that the first set of features possess a larger discriminative power for the task under consideration.

Another interesting application of AEs is shown in the work of Jia et al. (2016). Here, the nonlinear mapping implemented by deep AEs is exploited to pre-train an ANN which is in turn used to perform fault diagnosis both on rolling element bearings and planetary gearboxes. More specifically, the weights between two hidden layers are initialized by training an AE to minimize the reconstruction error of the input values specified by the first hidden layer. With this pre-training strategy, the feature extraction ability of AEs is used to encode relevant properties of the data directly into the ANN weight configuration.

AE architectures can also be used to estimate a health indicator which measures the “distance” of a test data point to the training healthy class (Michau et al., 2017; Michau et al., 2019; Wen and Gao, 2018). For example, in the work of Michau et al. (2019) a system comprising of an AE and a one class-classifier is trained with only healthy data to assess the health state of a complex electricity production plant. In this work, both AE and one-class classifier have the structure of a particular type of neural network called Extreme Learning Machine (ELM). ELM-based AEs have been also successfully employed in Michau et al. (2017) and Yang et al. (2016), among others.

3.2.2.2 Convolutional Neural Networks

CNNs are particularly advantageous in the context of fault diagnosis since they implement the feature extraction and classification tasks in an end-to-end fashion. Moreover, they can be applied to several data structures, including both time-series and images (Jiao et al., 2020). A common strategy to employ 2D-CNNs[6] in PHM applications is to feed these models with image-like data. This poses the problem of how to convert sensor measurements, which are typically in the form of multivariate time-series, into a grid-like structure. Examples of this procedure can be found, for example, in Ding and He (2017), Sun et al. (2017), Guo et al. (2018b), Wen et al. (2018), Cao et al. (2019), Islam and Kim (2019a), Li et al. (2019a), Wang et al. (2019). Most of these works employ popular signal processing techniques to perform the two-dimensional mapping. In particular, Li et al. (2019a) use the S-transform to map bearing vibrational data into a time-frequency representation. Similarly, in Ding and He (2017), Sun et al. (2017), Guo et al. (2018b), Cao et al. (2019), Islam and Kim (2019a) transformations based on the wavelet transform are used to process data gathered from bearings, rotating machinery and gears. An additional strategy is proposed in Wen et al. (2018), where the following mapping is applied to convert time-series data into two-dimensional images:

$$P(j,k) = \text{round}\left\{\frac{L\left(\left(j-1\right)\times M+k\right)-\text{Min}\left(L\right)}{\text{Max}\left(L\right)-\text{Min}\left(L\right)}\times 255\right\}, \quad (9)$$

[6]We use the notation “(1D)2D-CNN” to indicate a CNN architecture with (one) two-dimensional filters.

where the input signal is a vector of size M^2, $L(j)$ is signal magnitude at the jth time step and $P(j,k)$ is the intensity of the (j,k) pixels in the output image. This technique has been applied to data extracted from rolling element bearings and hydraulic and centrifugal pumps resulting in nearly optimal fault classification accuracy in all three cases.

Another class of methods applies CNNs directly to image data, thus leveraging the great success of these architectures in CV tasks. For example, Janssens et al. (2018); Jia et al. (2019) use CNNs to perform fault diagnosis of rotating machinery based on infrared thermal videos and images respectively. Yuan et al. (2018) propose a method that fuses features extracted from different data structures, including infrared images, for CNN-based fault classification of a rotor system.

Alternatively to 2D-CNNs, 1D-CNNs can be used to directly process time-series data. The literature contains a large number of examples that propose to apply 1D-CNN to bearing (Eren, 2017; Chen et al., 2018; Eren et al., 2019; Qin et al., 2019; Xueyi et al., 2019) and gears (Jing et al., 2017; Yao et al., 2018; Han et al., 2019b) fault diagnosis. Chen et al. (2018), for instance, propose a novel DL model, based on the popular Inception architecture (Szegedy et al., 2015) and a particular type of dilated convolution (Holschneider et al., 1990). The model is trained with data generated from artificial bearing damages and achieves very good performances on real data. The proposed method is pre-processing-free since it takes as input raw temporal signals directly.

The ability of CNN architectures to extract features in an end-to-end manner is tested in Jing et al. (2017). Here, the authors compare the quality of these features with a number of benchmarks consisting of conventional feature engineering approaches. The results show the superiority of the feature-learning pipeline implemented by CNNs over manual feature extraction.

Finally, CNN have also been applied to generate health indicators and to estimate the degradation trend of rolling bearings (Guo et al., 2018a; Yoo and Baek, 2018). In Yoo and Baek (2018), for instance, the authors apply a continuous wavelet transform to the data and feed the resulting two-dimensional images into a 2D-CNN which, in turn, outputs the health indicator.

3.2.2.3 Recurrent Neural Networks

RNNs have been mainly used for fault prognosis and only a relatively small number of works focus on their application to fault diagnosis. Some examples are (Li et al., 2018a; Li et al., 2018b Qiu et al., 2019) for bearings (Zhao H. et al., 2018; Zhao Q. et al., 2018 Yuan and Tian, 2019), for chemical processes control [see Tenessee Eastman dataset (Chen, 2019)] and (Lei et al., 2019) for wind turbines.

These methods can be divided into two categories: "RNN + classifier" and end-to-end approaches. The works of Li et al. (2018a, 2018b) and Yuan and Tian (2019) belong to the first category. The first employs an LSTM-based architecture to extract informative features from the input data. The so-obtained features are then fed into a softmax classifier that performs fault classification. Yuan and Tian (2019) use a GRU network to obtain dynamic features from several sub-sequences extracted from the raw signals. Multi-class classification is performed by a final softmax layer fed with the features obtained by the GRU module.

Zhao H. et al., 2018; Zhao Q. et al., 2018 Qiu et al. (2019); Lei et al. (2019) use RNN architecture in an end-to-end manner. For instance, Qiu et al. (2019) use a variant of Bi-LSTMs specifically designed to process long-term dependencies, to directly classify fault types. The network is trained with a set of features extracted by means of wavelet packet transform and employs softsign activation functions to contrast the vanishing gradient problem. Another end-to-end approach is proposed in Lei et al. (2019) where the authors use an LSTM-based model for fault diagnosis of a wind turbine. In this work, features are directly extracted by the network and there is no need for manual feature extraction. The proposed method is shown to outperform existing fault diagnosis techniques, such as ANNs, SVMs and CNNs.

3.2.2.4 Hybrid

With hybrid approaches we mean all those methods that combine the benefits provided by AEs, CNNs and RNNs models into single powerful systems.

For example, Li et al. (2019d); Park et al. (2019) propose techniques leveraging the efficacy of AEs in extracting valuable features and the advantages provided by RNN-architectures in analyzing time-dependent data. In Li et al. (2019d), first stacked AEs generate a latent representation of the raw input rotary machinery data. An LSTM network is then used to predict the value corresponding to the 10-th time step in the feature sequence given the previous 9. The reconstruction error between prediction and ground truth value is used to determine if the datum is anomalous or not.

An alternative approach consists in using recurrent models in the form of AEs to better deal with time-series data. In Liu et al. (2018), for instance, a GRU-based DAE is proposed for rolling bearing fault diagnosis. Specifically, the proposed GRU model is used to predict the next period given the previous one. As many such models as the number of faults are trained and classification is performed by selecting the model providing the lowest reconstruction error.

CNN-based architectures can also be combined with other types of networks for the purpose of fault diagnosis. In Liu et al. (2019b), for instance, a one-dimensional convolutional-DAE is proposed to extract features from bearing and gearbox data. This model is given corrupted time-series as input and its goal is to clean and reconstruct them at the output level. The so-learned features are then fed into an additional CNN model that performs the classification task.

In Zhao et al. (2017), Pan et al. (2018), Xueyi et al. (2019), the combination of CNNs and RNNs is investigated. For example, in Xueyi et al. (2019) a 1D-CNN and a GRU network are used to extract discriminative features from acoustic and vibration signals respectively. The so-obtained features are then concatenated and fed into a softmax classifier which performs gear pitting fault diagnosis. This hybrid method is shown to outperform CNN and GRU applied individually to the same data.

Pan et al. (2018), instead, proposes a method fusing a 1D-CNN and an LSTM network into a single structure. The LSTM takes as input the output of the CNN and performs fault diagnosis over bearing data. The proposed algorithm provides nearly optimal performances on the test set.

3.2.3 Prognosis

3.2.3.1 Autoencoder

AEs are typically used in combination with other regression techniques for the purpose of fault prognosis. The literature contains examples of AE-based techniques applied to RUL estimation of bearings (Ren et al., 2018; Xia et al., 2019), machining centers (Yan et al., 2018), aircraft engines (Ma et al., 2018) and lithium-ion batteries (Ren et al., 2018b). The role of AEs in all the above references is to perform automatic feature extraction to facilitate the work of regression or classification methods used for health state assessment or RUL estimation. Xia et al. (2019), for example, utilize a DAE and a softmax classifier trained on top of the AE embedding to classify the inputs into different degradation stages. Then, ANN-based regressors are used to model each stage separately. The final RUL is obtained by applying a smoothing operation to all the previously computed regression models.

In Ma et al. (2018), AEs are used in a similar manner. The authors propose a system composed of a DAE, a SAE and a logistic regressor to predict the RUL on an aircraft engine. The first AE module generates low-level features which are in turn fed into the second AE model which outputs a new set of high-level features. Finally, the logistic regressor predicts the RUL based on the features extracted by the second AE.

3.2.3.2 Convolutional Neural Networks

CNN architectures have been extensively explored also for fault prognosis. These methods have been mainly applied to open-source evaluation platforms such as the popular NASA's C-MAPSS dataset (Saxena and Goebel, 2008) for aero-engine unit prognostics (Babu et al., 2016; Li et al., 2018a; Li et al., 2018b Wen et al., 2019a) and the PRONOSTIA dataset (Ali et al., 2015) for bearings health assessment (Ren et al., 2018a; Zhu et al., 2018; Li et al., 2019c; Wang et al., 2019b; Yang et al., 2019).

In Li et al., 2018a; Li et al., 2018b a 1D-CNN model is used to predict the RUL on the C-MAPSS dataset. Data are first chunked in fixed-length windows and then directly fed into the network without any pre-processing step. Despite the relative simplicity of the employed architecture, the proposed technique is able to provide pretty good prediction results, especially in proximity of the final failure.

In Wen et al. (2019a) the authors build upon the work of Li et al., 2018a; Li et al., 2018b and propose a novel CNN model for RUL estimation which draws inspiration from the popular ResNet architecture (He et al., 2016). The proposed technique is shown to outperform traditional methods such as SVMs, ANNs, LSTM and the model proposed by Li et al., 2018a; Li et al., 2018b in terms on RUL mean and standard deviation on the C-MAPSS dataset.

In the context of bearing fault prognosis, Ren et al. (2018a) propose a new approach based on manual feature extraction and CNNs for RUL estimation. First, a new method for feature extraction is proposed to generate a feature map which is highly correlated with the decay of bearing vibration over time. This feature map is then fed into a deep 2D-CNN which outputs the RUL estimate. Linear regression is then used as a smoothing method to reduce the discontinuity problem in the final prediction result. Experiments show that the proposed method is able to provide improved prediction accuracy in bearing RUL estimation.

3.2.3.3 Recurrent Neural Networks

The application of RNN architectures to fault prognosis have been explored on various industrial components such that lithium-ion-batteries (Zhang et al., 2018), gears (Xiang et al., 2020), fuel cells (Liu et al., 2019a), and on the C-MAPSS dataset (Yuan et al., 2016; Zheng et al., 2017; Wu et al., 2018a; Wu et al., 2018b; Chen et al., 2019; Elsheikh et al., 2019; Wu et al., 2020). One of the most popular RNN-based approaches proposed in the literature is the work of Wu et al. (2018b). The authors first extract dynamic features containing inter-frame information and then use these features to train a vanilla-LSTM model to predict the RUL. An SVM model is employed to detect the degradation starting point. The proposed technique is shown to consistently outperform a standard RNN and a GRU model trained on the same dataset. The remarkable performances of LSTM networks on the RUL estimation task are further confirmed by the work of Zheng et al. (2017). The authors combine LSTM layers with a feed-forward neural network, showing that the proposed approach provides better performances than ANNs, SVM and CNNs. In Xiang et al. (2020), the attention mechanism is used to enhance the performances of an LSTM network on the prediction of the RUL on gears. The aforementioned model, named LSTMP-A, is trained with time-domain and frequency-domain features and its comparison with other recurrent models shows that it provides the best prediction accuracy.

3.2.3.4 Hybrid

Hybrid approaches have been also applied in the context of fault prognosis. For instance, the literature contains examples of AE + RNN (Lal Senanayaka et al., 2018; Deng et al., 2019) and CNN + RNN (Zhao et al., 2017; Mao et al., 2018; Li et al., 2019b) combinations. In Zhao et al. (2017) sensory data from milling machine cutters are processed by a novel technique combining a CNN component and an LSTM network. The CNN is used to extract local features, whereas a bi-LSTM captures long-term dependencies and take into account both past and future contexts. A sequence of fully connected layers and a linear regression layer takes as input the output of the LSTM and predicts the tool-wear level.

Similarly, Mao et al. (2018) combine LSTM and CNN models for feature extraction and RUL prediction. In particular, time-series from the C-MAPPS dataset are first sliced by applying a time-window. The resulting data are then independently fed into an LSTM network and a CNN. The features extracted by these two networks are then combined and further processed by an additional LSTM network and a fully connected layer which predicts the RUL.

Deng et al. (2019) propose a method based on the combination of stacked SAEs and a GRU model. The AE is used for automatic feature extraction and the GRU is used to model the mapping from the features extracted by the AE to the RUL values. The proposed method is applied to the C-MAPPS dataset, showing satisfactory results.

3.2.4 Discussion

3.2.4.1 Dependency on Feature Extraction

One of the key advantages of DL algorithms over traditional ML approaches stands in their lower degree of dependence on the feature extraction step. Their input can consist of either raw data or a set of manually extracted features, depending on the amount of prior information available to the user about the task under consideration.

3.2.4.2 Model Selection

As already discussed for traditional ML algorithms, a universal approach valid for all possible application scenarios does not exist. In general, the nature of the problem dictates which method to utilize. For instance, when the PHM problem at hand involves image data, the usage of 2D CNN might be preferred. On the other hand, when sensor measurements consisting of time-series data have to be analyzed, 1D CNN and RNN architectures are more sensible choices. Ultimately, the final model can be selected by evaluating each candidate on the same metrics mentioned at the end of paragraph 3.1.3.2 and comparing the corresponding scores.

3.2.4.3 Overfitting

As already mentioned before, a larger number of hidden layers is often associated with a higher risk of overfitting. Beyond the techniques already discussed for ANNs (e.g., cross-validation, early-stopping and regularization), deep models can be equipped with more advanced tools to contrast over-training. A popular example is the Dropout technique (Srivastava et al., 2014) which randomly drops neurons from the neural network at training time. Intuitively, this prevents the network to specialize on a particular set of data. Dropout is used, for instance, in Han et al. (2019b) and Wang et al. (2019) with the corresponding parameter fixed at 0.5. Finally, data augmentation can be also used to generate new images by applying simple transformations (e.g., rotation, mirroring, cropping, padding) to the training data. For instance, this technique is applied in Wang et al. (2019) to time-frequency images obtained from bearing accelerometers, in order to increase the size and the level of diversity of the training set.

4 CRITIQUE AND FUTURE DIRECTIONS

In the previous section, we have discussed some of the most popular DL techniques that have been applied to PHM problems over the last few years. We have compared traditional ML approaches with DL techniques, trying to highlight the strengths of both methods and emphasizing the change of paradigm introduced by the so-called DL revolution.

The goal of this section is to shed some light over a number of open challenges that need to be addressed to bridge the gap between research and industrial applications. We start by briefly discussing some of these open questions and some limitations of DL models that hinder their solution. Then, we discuss some first attempts to cope with these challenges along with some proposals of future investigations. Our goal is to provide the reader with a set of possible fruitful research directions that we consider as valuable candidates to further increase the impact of DL to PHM.

4.1 Open Challenges

4.1.1 Reliability and Interpretability

One of the most common criticisms to DL models arises from their black-box nature, i.e., the sometimes opaque mechanism by which they make their decisions. This characteristic of deep models derives from one of the properties that allows them to successfully tackle several different tasks: the complex sequence of nonlinear operations they implement across their deep architectures. A complete mathematical characterization of the behavior of DL models in light of their inherent complexity is very hard to obtain. This negative property of deep networks represents a significant limitation to their deployment in areas such as healthcare, finance, and PM. In these delicate contexts, humans need to have control over their tools and it is not always possible to sacrifice trust and transparency for better performances. It is therefore urgent to enhance the level of interpretability of these models in order to make them fully deployable while minimizing the risks.

However, it is not straightforward to provide a unique definition of the concept of interpretability (Lipton, 2018). DL models can be, for instance, enhanced with complementary functionalities responsible for providing a post-hoc explanation of their actions. Alternatively, one can build some notion of interpretability directly into the models in order to constrain their learning process to align with some inductive biases that we might deem trustworthy. The strategy of providing post-hoc explanations of the model behavior have been widely investigated in CV (Ribeiro et al., 2016; Zhou et al., 2016; Lundberg and Lee, 2017). Few attempts, however, have been made to extend these approaches to time-series data [see for example (Fawaz et al., 2019), (Guillemé et al., 2019)].

Imposing appropriate inductive biases on DL models have been recently identified as a key step to perform unsupervized learning tasks (Locatello et al., 2019a; Locatello et al., 2019b). Some possible inductive biases can derive from a-priori available physical knowledge of the problem under consideration. This complementary information can be incorporated directly into the network architecture or can be used to drive a model toward more meaningful output decisions. We discuss some of these approaches later in this section.

To conclude this discussion, it is worth mentioning that another important requirement for interpretable and transparent models stands in their ability to provide uncertainty estimates about their predictions. Uncertainty can derive both from the intrinsic stochasticity of the task (aleatoric uncertainty) and from the approximations introduced by our imperfect model (parametric uncertainty). Bayesian approaches can in principle deal with uncertainty estimation and their combination with DL methods is a hot research area (Damianou and Lawrence, 2013; Blundell et al., 2015; Garnelo et al., 2018).

4.1.2 Highly Specialized Models

An increasing amount of experimental evidence (Zhang et al., 2017; Beery et al., 2018; Arjovsky et al., 2019) has recently attracted the attention of the scientific community on an additional relevant limitation of deep models: they often tend to learn "shortcuts" instead of the underlying physical mechanisms describing the data. For instance, let's consider the task of classifying cows and camels based on a training set containing labeled images where cows are mostly found in green pastures and camels in sandy deserts (Beery et al., 2018). Testing our model on images of cows taken in a different environment, such as beaches, leads to a wrong classification decision. Similar generalization deficiencies can be also observed in the context of PHM applications. Typically, labeled data are available only for a single machine; training a model on these data can lead to good performances on a test set extracted from the same machine but to very disappointing results on a similar machine operating at slightly different operating conditions. The variability in the machines' operational modes can arise from differences in specific choices in their design, or to external factors (e.g., environmental variables such as humidity, temperature, seasonality). Ideally, an efficient model should be able to deal with these factors of variability and provide predictions that are robust to changing operating conditions. On the other hand, the majority of the DL approaches proposed in the literature do not address this point and focus on relatively narrow systems without taking generalization into account. If we really aim at designing "Intelligent" systems that can take decisions following similar cognitive patterns as those characterizing human decision making, we have to provide new solutions to the aforementioned shortcomings.

4.1.3 Data Scarcity

An immediate consequence of using DL models is that, by increasing the depth of the network, the number of parameters associated with it grows accordingly. As a result, finding an optimal weight configuration requires training these networks with very large datasets. In particular, supervised learning approaches are based on the availability of large numbers of labeled data instances for each class under consideration. This aspect poses a significant practical limitation on the application of DL models to the industry domain. In the case of fault diagnosis, for example, it is difficult to find an adequately large number of data for each possible fault. This is mainly because, luckily, faulty data tend to be relatively rare compared to healthy ones. Furthermore, it might also be the case that some faults are not even a-priori known and it is, therefore, impossible to precisely characterize them. This lack of representativeness (Michau et al., 2018) of the training data delineates a very common scenario in practical applications. Two possible alternative approaches can be adopted to cope with it: the first is to design algorithms that are less data-intensive, whereas the second is to generate artificial data that strongly resemble real ones. We discuss some of these methods in the next section.

4.2 Possible Solutions

4.2.1 Fusing Deep Learning With Physics

One possible way to cope with the aforementioned challenges is to incorporate information about the physics of the system under consideration into the learning process. DL algorithms, in and of themselves, are not able to capture the primitive causal mechanisms at the basis of the input observations (Pearl, 2019). On the other hand, physical models of complex systems are built from fundamental laws of physics but often rely on relatively strong approximations which result in poor predictive power. Taking prior physics knowledge into account can be helpful in inducing a higher level of interpretability into deep models and in improving their generalization performances. Hybrid models integrating the flexibility of modern data-driven techniques and the transparency of physics models have the potential of overcoming the limitations of the two stand-alone approaches by exploiting their individual strengths.

In the context of PHM, a relatively small number of works have been proposed in this direction. For example, in Chao et al. (2019), a high-fidelity performance model of an aircraft engine is first calibrated on real data by using an Unscented Kalman Filter (Julier and Uhlmann, 1997) and then used to generate unobserved physical quantities that are in turn employed to enhance the input space of a DL model. The results show that the new input space including both observed and virtual measurements contributes in significantly improving the performances of the model.

An alternative way to fuse physics knowledge and data-driven methods is described in Dourado and Viana (2020) and Nascimento and Viana (2019). In these works, well-known physics-based cumulative damage models are complemented by data-driven techniques whose goal is to explain some additional phenomena that the original model is not able to accurately describe. The final model has a sound physical interpretation and provides refinements over the original physics model thanks to its data-driven component.

We conclude this part by noticing that physics knowledge could also be incorporated into deep models directly at the architecture level. Recent research in Graph Neural Networks (Sanchez-Gonzalez et al., 2018; Cranmer et al., (2020)) shows that these kind of models are particularly suitable to encode and exploit prior physics knowledge, for instance, given in the form of Partial Differential Equations over space and time. An example of an industrial application of these models is provided by Park and Park (2019) who use a specific type of GNN to estimate the power generated by a wind farm by modeling the physics interactions between the individual turbines.

4.2.2 Domain Adaptation

The high variability of machines' operating conditions and the problem of data scarcity motivate the introduction of techniques capable of transferring the knowledge gained from a well-known machine to another for which data are not as abundant. Transfer Learning (TL) is a class of ML methods whose goal is to address this problem. Traditional TL approaches (Yosinski et al., 2014) are based on the following rationale: first, a deep network is trained on a large dataset to perform a specific task. Then, the same network is used to perform a similar task simply by fine-tuning its final layers on a few instances from the new dataset. Recent works in the context of fault diagnosis and fault prognosis have successfully applied this idea on datasets from induction motors (Shao et al., 2019a; Shao et al., 2019b), gearboxes (Cao et al., 2018; He et al., 2019; Shao et al., 2019a; Shao et al., 2019b), bearings (Shao et al., 2019a; Shao et al., 2019b Wen et al., 2019b) and centrifugal pumps (Wen et al., 2019b).

Besides traditional TF methods, unsupervized Domain Adaptation (DA) techniques have also been recently applied to PHM tasks. DA is a sub-field of TF, whose goal is to maximize the performances on the target domain for which only few unlabeled data are available by exploiting a labeled data from the so-called source domain. The two domains are commonly assumed to share similar features even though a model trained on the source domain will usually provide poor performances on the target domain. This is typically due to a distributional shift between the marginal distributions describing the two sets of data. DA techniques have witnessed an increasing attention since the introduction of the so-called adversarial DA methods (Ganin and Lempitsky, 2014; Ganin et al., 2016; Tzeng et al., 2017). These approaches draw inspiration from the training procedure used by the popular Generative Adversarial Networks (GANs) (Goodfellow et al., 2014) to efficiently align source and target domain features in a common latent-space. Several new techniques (Han et al., 2019a; Wang et al., 2019a; Wang and Liu, 2020) based on this class of DA approaches have been recently proposed in the PHM literature. Other references on DA and TF approaches in the context of fault diagnosis can be found in the recent review works of Li et al. (2020) and Zheng H. et al., 2019; Zheng Z. et al., 2019.

4.2.3 Artificial Data Generation

Generative models such as GANs and VAEs have achieved impressive results in generating photo-realistic artificial data in the context of CV. However, the task of generating realistic problem-specific time-series data is still relatively unexplored compared to artificial image generation. Unsurprisingly, existing approaches in this context make large use of GANs. In Donahue et al. (2018), for instance, GANs are used for music and speech synthesis. In Nik Aznan et al., (2019), Haradal et al. (2018), and Hyland et al., (2017) the authors propose new GAN-based methods that generate medical data such as electroencephalographic (EEG) brain signals, and time-dependent health parameters of patients hospitalized in the Intensive Care Unit (ICU). The recent method proposed by Yoon et al. (2019) provides new state-of-the-art performance for realistic time-series generation.

The benefits of such approaches in the context of PHM could be significant. One of their most direct application is to perform data augmentation in order to tackle to problem of lack of representativeness and therefore improving the performance of data-intensive DL models. To the authors' knowledge, only a small number of works have started exploring this idea and some first interesting results have already been produced (Mao et al., 2019; Shao et al., 2019a; Shao et al., 2019b Wang et al., 2019).

5 DISCUSSION

PM, as a key player in the Industry 4.0 paradigm, strongly relies on some of the most recent advances in hardware technology, communication systems and data science. Among them, DL techniques have gained popularity over the last few years in light of their excellent performances in processing complex data in an end-to-end fashion. In this review, we have described several applications of these methods to PHM. In particular, we have discussed the advantages they introduce over traditional ML techniques, stressing on their improved representational power and their ability to automatically extract informative features from data. Despite its great success, DL presents some shortcomings that limit its large-scale deployment in industrial applications. Its low level of interpretability, its generalization deficiencies and its data-intensive nature are some of the main weaknesses DL needs to overcome to close the gap between academia and industrial deployment. In this review, we identified three research areas that we believe could address or alleviate the aforementioned open challenges, namely: physics-enhanced techniques, domain adaptation and artificial data generation. The first aims to improve interpretability by grounding data-driven methods on well-understood physics models of the system under consideration. Furthermore, incorporating prior physics knowledge into DL algorithms can be seen as imposing meaningful inductive biases into the learning process, resulting in improved generalization and reasoning. Domain adaptation provides a set of tools to transfer the knowledge acquired on a well-known industrial component to other similar assets for which data are less abundant. Finally, artificial data generation techniques can be used to cope with the lack of representativeness problem and the data-intensive nature of DL algorithms. Some of these lines of research have already shown interesting results, while others, although very promising, are only in their infancy.

AUTHOR CONTRIBUTIONS

LB designed the study and wrote the manuscript. IK contributed to the final version of the manuscript and supervised the project.

REFERENCES

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., et al. (2016). Tensorflow: large-scale machine learning on heterogeneous distributed systems. *arXiv*.

Abbasion, S., Rafsanjani, A., Farshidianfar, A., and Irani, N. (2007). Rolling element bearings multi-fault classification based on the wavelet denoising and support vector machine. *Mech. Syst. Signal Process.* 21, 2933–2945. doi:10.1016/j.ymssp.2007.02.003

Abdallah, I., Dertimanis, V., Mylonas, H., Tatsis, K., Chatzi, E., Dervili, N., et al. (2018). "Fault diagnosis of wind turbine structures using decision tree learning algorithms with big data," in 28th European Safety and Reliability Conference (ESREL 2018), Trondheim, Norway, June 17–21, 2018, 3053–3061. doi:10.1201/9781351174664-382

Abu-Mahfouz, I. A. (2005). A comparative study of three artificial neural networks for the detection and classification of gear faults. *Int. J. Gen. Syst.* 34, 261–277. doi:10.1080/03081070500065726

Ali, J. B., Fnaiech, N., Saidi, L., Chebel-Morello, B., and Fnaiech, F. (2015). Application of empirical mode decomposition and artificial neural network for automatic bearing fault diagnosis based on vibration signals. *Appl. Acoust.* 89, 16–27. doi:10.1016/j.apacoust.2014.08.016

Appana, D. K., Islam, M. R., and Kim, J.-M. (2017). "Reliable fault diagnosis of bearings using distance and density similarity on an enhanced k-nn," in *Australasian conference on artificial life and computational intelligence*. Editors M. Wagner, X. Li, and T. Hendtlass (Cham, Switzerland: Springer), 193–203.

Arjovsky, M., Bottou, L., Gulrajani, I., and Lopez-Paz, D. (2019). Invariant risk minimization. *arXiv*.

Ayhan, B., Chow, M.-Y., and Song, M.-H. (2006). Multiple discriminant analysis and neural-network-based monolith and partition fault-detection schemes for broken rotor bar in induction motors. *IEEE Trans. Ind. Electron.* 53, 1298–1308. doi:10.1109/tie.2006.878301

Babu, G. S., Zhao, P., and Li, X.-L. (2016). "Deep convolutional neural network based regression approach for estimation of remaining useful life," in *International conference on database systems for advanced applications*. Editors S. Navathe, W. Wu, S. Shekhar, X. Du, X. Wang, and H. Xiong.(New York, NY: Springer), 214–228.

Bashar, M. A., Nayak, R., and Suzor, N. (2020). Regularising lstm classifier by transfer learning for detecting misogynistic tweets with small training set. *Knowl. Inf. Syst.* 62 (10), 4029–4054. doi:10.1007/1074 s10115-020-01481-0

Bay, H., Ess, A., Tuytelaars, T., and Van Gool, L. (2008). Speeded-up robust features (surf). *Comput. Vis. Image Understand.* 110, 346–359. doi:10.1016/j.cviu.2007.09.014

Beery, S., Van Horn, G., and Perona, P. (2018). "Recognition in terra incognita," in Proceedings of the European Conference on Computer Vision (ECCV), 2018, 456–473.

Bengio, Y., Simard, P., and Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Trans. Neural Network.* 5, 157–166. doi:10.1109/72.279181

Benkercha, R., and Moulahoum, S. (2018). Fault detection and diagnosis based on c4.5 decision tree algorithm for grid connected pv system. *Sol. Energy* 173, 610–634. doi:10.1016/j.solener.2018.07.089

Blundell, C., Cornebise, J., Kavukcuoglu, K., and Wierstra, D. (2015). Weight uncertainty in neural networks. *arXiv*.

Bruckner, D., Stanica, M.-P., Blair, R., Schriegel, S., Kehrer, S., Seewald, M., et al., (2019). An introduction to opc ua tsn for industrial communication systems. *Proc. IEEE* 107, 1121–1131. doi:10.1109/jproc.2018.2888703

Cao, P., Zhang, S., and Tang, J. (2018). Preprocessing-free gear fault diagnosis using small datasets with deep convolutional neural network-based transfer learning. *IEEE Access* 6, 26241–26253. doi:10.1109/access.2018.2837621

Cao, X.-C., Chen, B.-Q., Yao, B., and He, W.-P. (2019). Combining translation-invariant wavelet frames and convolutional neural network for intelligent tool wear state identification. *Comput. Ind.* 106, 71–84. doi:10.1016/j.compind.2018.12.018

Chao, M. A., Kulkarni, C., Goebel, K., and Fink, O. (2019). Hybrid deep fault detection and isolation: combining deep neural networks and system performance models. *arXiv*.

Chen, J., Jing, H., Chang, Y., and Liu, Q. (2019). Gated recurrent unit based recurrent neural network for remaining useful life prediction of nonlinear deterioration process. *Reliab. Eng. Syst. Saf.* 185, 372–382. doi:10.1016/j.ress.2019.01.006

Chen, X. (2019). [Dataset] Tennessee eastman simulation dataset. doi:10.21227/4519-z50210.1037/t72896-000

Chen, X., Shen, Z., He, Z., Sun, C., and Liu, Z. (2013). Remaining life prognostics of rolling bearing based on relative features and multivariable support vector machine. *Proc. IME C J. Mech. Eng. Sci.* 227, 2849–2860. doi:10.1177/0954406212474395

Chen, Y., Peng, G., Xie, C., Zhang, W., Li, C., and Liu, S. (2018). Acdin: bridging the gap between artificial and real bearing damages for bearing fault diagnosis. *Neurocomputing* 294, 61–71. doi:10.1016/j.neucom.2018.03.014

Chine, W., Mellit, A., Lughi, V., Malek, A., Sulligoi, G., and Massi Pavan, A. (2016). A novel fault diagnosis technique for photovoltaic systems based on artificial neural networks. *Renew. Energy* 90, 501–512. doi:10.1016/j.renene.2016.01.036

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., et al. (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv*. doi:10.3115/v1/d14-1179

Cranmer, M., Greydanus, S., Hoyer, S., Battaglia, P., Spergel, D., and Ho, S. (2020). "Lagrangian neural networks," in ICLR 2020 workshop on integration of deep neural models and differential equations.

Damianou, A., and Lawrence, N. (2013). "Deep Gaussian processes," in Proceedings of the Sixteenth International Conference on Artificial intelligence and statistics, AISTATS 2013, Scottsdale, AZ, April 29–May 1, 2013, 207–215.

Davis, S., and Mermelstein, P. (1980). Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Trans. Acoust. Speech Signal Process.* 28, 357–366. doi:10.1109/tassp.1980.1163420

Deng, K., Zhang, X., Cheng, Y., Zheng, Z., Jiang, F., Liu, W., et al. (2019). "A remaining useful life prediction method with automatic feature extraction for aircraft engines," in 2019 18th IEEE international conference on trust, security and privacy in computing and communications/13th IEEE international conference on big data science and engineering (TrustCom/BigDataSE), Rotorua, New Zealand, August 5–8, 2019 (IEEE), 686–692.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: pre-training of deep bidirectional transformers for language understanding. *arXiv*.

Ding, X., and He, Q. (2017). Energy-fluctuated multiscale feature learning with deep convnet for intelligent spindle bearing fault diagnosis. *IEEE Trans. Instrum. Meas.* 66, 1926–1935. doi:10.1109/tim.2017.2674738

Donahue, C., McAuley, J., and Puckette, M. (2018). Adversarial audio synthesis. *arXiv*.

Dong, S., Luo, T., Zhong, L., Chen, L., and Xu, X. (2017). Fault diagnosis of bearing based on the kernel principal component analysis and optimized k-nearest neighbor model. *J. Low Freq. Noise Vib. Act. Contr.* 36, 354–365. doi:10.1177/1461348417744302

Dourado, A., and Viana, F. A. (2020). Physics-informed neural networks for missing physics estimation in cumulative damage models: a case study in corrosion fatigue. *J. Comput. Inf. Sci. Eng.* 20, 061007. doi:10.1115/1.4047173.

Elforjani, M., and Shanbr, S. (2018). Prognosis of bearing acoustic emission signals using supervised machine learning. *IEEE Trans. Ind. Electron.* 65, 5864–5871. doi:10.1109/tie.2017.2767551

Elsheikh, A., Yacout, S., and Ouali, M.-S. (2019). Bidirectional handshaking lstm for remaining useful life prediction. *Neurocomputing* 323, 148–156. doi:10.1016/j.neucom.2018.09.076

Eren, L. (2017). Bearing fault detection by one-dimensional convolutional neural networks. *Math. Probl. Eng.*, 2017, 1–9. doi:10.1155/2017/8617315

Eren, L., Ince, T., and Kiranyaz, S. (2019). A generic intelligent bearing fault diagnosis system using compact adaptive 1d cnn classifier. *J Sign Process Syst* 91, 179–189. doi:10.1007/s11265-018-1378-3

Fawaz, H. I., Forestier, G., Weber, J., Idoumghar, L., and Muller, P.-A. (2019). Deep learning for time series classification: a review. *Data Min. Knowl. Discov.* 33, 917–963

Fernández-Francos, D., Martínez-Rego, D., Fontenla-Romero, O., and Alonso-Betanzos, A. (2013). Automatic bearing fault diagnosis based on one-class ν-SVM. *Comput. Ind. Eng.* 64, 357–365. doi:10.1016/j.cie.2012.10.013

Fink, O. (2020). "Data-driven intelligent predictive maintenance of industrial assets," in *Women in industrial and systems engineering.* Editor A. Smith (New York, NY: Springer), 589–605

Friedman, J. H. (1987). Exploratory projection pursuit. *J. Am. Stat. Assoc.* 82, 249–266. doi:10.1080/01621459.1987.10478427

Ganin, Y., and Lempitsky, V. (2014). Unsupervised domain adaptation by backpropagation. *arXiv*.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., et al. (2016). Domain-adversarial training of neural networks. *J. Mach. Learn. Res.* 17, 2096–2030.

Garnelo, M., Schwarz, J., Rosenbaum, D., Viola, F., Rezende, D. J., Eslami, S., et al. (2018). Neural processes. *arXiv*.

Gebraeel, N., Lawley, M., Liu, R., and Parmeshwaran, V. (2004). Residual life predictions from vibration-based degradation signals: a neural network approach. *IEEE Trans. Ind. Electron.* 51, 694–700. doi:10.1109/tie.2004.824875

Gharavian, M. H., Almas Ganj, F., Ohadi, A. R., and Heidari Bafroui, H. (2013). Comparison of fda-based and pca-based features in fault diagnosis of automobile gearboxes. *Neurocomputing* 121, 150–159. doi:10.1016/j.neucom.2013.04.033

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., et al. (2014). "Generative adversarial nets," in Proceedings of the 27th International Conference on Advances in neural information processing systems, Montreal, Canada, June 10, 2014, 2672–2680.

Gryllias, K. C., and Antoniadis, I. A. (2012). A support vector machine approach based on physical model training for rolling element bearing fault detection in industrial environments. *Eng. Appl. Artif. Intell.* 25, 326–344. doi:10.1016/j.engappai.2011.09.010

Guillemé, M., Masson, V., Rozé, L., and Termier, A. (2019). "Agnostic local explanation for time series classification," in 2019 IEEE 31st international conference on tools with artificial intelligence (ICTAI), Portland, OR, November 2019, 432–439.

Guo, L., Lei, Y., Li, N., Yan, T., and Li, N. (2018a). Machinery health indicator construction based on convolutional neural networks considering trend burr. *Neurocomputing* 292, 142–150. doi:10.1016/j.neucom.2018.02.083

Guo, S., Yang, T., Gao, W., and Zhang, C. (2018b). A novel fault diagnosis method for rotating machinery based on a convolutional neural network. *Sensors* 18, 1429. doi:10.3390/s18051429

Guyon, I., Gunn, S., Nikravesh, M., and Zadeh, L. A. (2006). *Feature extraction: foundations and applications (studies in fuzziness and soft computing)*. Berlin, Heidelberg: Springer-Verlag.

Han, T., Liu, C., Yang, W., and Jiang, D. (2019a). A novel adversarial learning framework in deep convolutional neural network for intelligent diagnosis of mechanical faults. *Knowl. Base Syst.* 165, 474–487. doi:10.1016/j.knosys.2018.12.019

Han, Y., Tang, B., and Deng, L. (2019b). An enhanced convolutional neural network with enlarged receptive fields for fault diagnosis of planetary gearboxes. *Comput. Ind.* 107, 50–58. doi:10.1016/j.compind.2019.01.012

Haradal, S., Hayashi, H., and Uchida, S. (2018). Biosignal data augmentation based on generative adversarial networks. *Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.* 2018, 368–371. doi:10.1109/EMBC.2018.8512396

Hastie, T., Tibshirani, R., and Friedman, J. (2001). *The elements of statistical learning. Springer series in 1206 statistics*. New York, NY: Springer New York Inc.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep residual learning for image recognition," in Proceedings of the IEEE conference on computer vision and pattern recognition, Las Vegas, NV, June 2016, 770–778.

He, Z., Shao, H., Zhang, X., Cheng, J., and Yang, Y. (2019). Improved deep transfer auto-encoder for fault diagnosis of gearbox under variable working conditions with small training samples. *IEEE Access* 7, 115368–115377. doi:10.1109/access.2019.2936243

Hess, A. (2002). "Prognostics, from the need to reality-from the fleet users and phm system designer/developers perspectives," in Proceedings, IEEE Aerospace Conference (IEEE), Big Sky, MT, March 9–16, 2002, vol. 6, 2791–2797.

Hochreiter, S., and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation* 9, 1735–1780. doi:10.1162/neco.1997.9.8.1735

Hofmann, T., Schölkopf, B., and Smola, A. J. (2008). Kernel methods in machine learning. *Ann. Stat.*, 36 , 1171–1220. doi:10.1214/009053607000000677

Holschneider, M., Kronland-Martinet, R., Morlet, J., and Tchamitchian, P. (1990). "A real-time algorithm for signal analysis with the help of the wavelet transform," in *Wavelets*. Editors J.-M. Combes, A. Grossmann, and P. Tchamitchian (Berlin, Heidelberg: Springer), 286–297.

Huang, H.-Z., Wang, H.-K., Li, Y.-F., Zhang, L., and Liu, Z. (2015). Support vector machine based estimation of remaining useful life: current research status and future trends. *J. Mech. Sci. Technol.* 29, 151–163. doi:10.1007/s12206-014-1222-z

Hubel, D. H., and Wiesel, T. N. (1968). Receptive fields and functional architecture of monkey striate cortex. *J. Physiol.* 195, 215–243. doi:10.1113/jphysiol.1968.sp008455

Hyland, S. L., Esteban, C., and Ra¨tsch, G. (2017). Real-valued (medical) time series generation with recurrent conditional gans. *Stat* 1050, 8.

Hyvärinen, A., and Oja, E. (2000). Independent component analysis: algorithms and applications. *Neural Networks* 13, 411–430. doi:10.1016/s0893-6080(00)00026-5

Ioffe, S., and Szegedy, C. (2015). Batch normalization: accelerating deep network training by reducing internal covariate shift. *arXiv*.

Islam, M. M. M., Kim, J., Khan, S. A., and Kim, J.-M. (2017). Reliable bearing fault diagnosis using bayesian inference-based multi-class support vector machines. *J. Acoust. Soc. Am.* 141, EL89. doi:10.1121/1.4976038

Islam, M. M. M., and Kim, J.-M. (2019a). Automated bearing fault diagnosis scheme using 2d representation of wavelet packet transform and deep convolutional neural network. *Comput. Ind.* 106, 142–153. doi:10.1016/j.compind.2019.01.008

Islam, M. M. M., and Kim, J.-M. (2019b). Reliable multiple combined fault diagnosis of bearings using heterogeneous feature models and multiclass support vector machines. *Reliab. Eng. Syst. Saf.* 184, 55–66.

Janssens, O., Van de Walle, R., Loccufier, M., and Van Hoecke, S. (2018). Deep learning for infrared thermal image based machine health monitoring. *IEEE ASME Trans. Mechatron.* 23, 151–159. doi:10.1109/tmech.2017.2722479

Jia, F., Lei, Y., Guo, L., Lin, J., and Xing, S. (2018). A neural network constructed by deep learning technique and its application to intelligent fault diagnosis of machines. *Neurocomputing* 272, 619–628. doi:10.1016/j.neucom.2017.07.032

Jia, F., Lei, Y., Lin, J., Zhou, X., and Lu, N. (2016). Deep neural networks: a promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data. *Mech. Syst. Signal Process.* 72-73, 303–315. doi:10.1016/j.ymssp.2015.10.025

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., et al. (2014). Caffe: convolutional architecture for fast feature embedding. *arXiv*.

Jia, Z., Liu, Z., Vong, C.-M., and Pecht, M. (2019). A rotating machinery fault diagnosis method based on feature learning of thermal images. *IEEE Access* 7, 12348–12359. doi:10.1109/access.2019.2893331

Jiao, J., Zhao, M., Lin, J., and Liang, K. (2020). A comprehensive review on convolutional neural network in machine fault diagnosis. *arXiv*.

Jing, L., Zhao, M., Li, P., and Xu, X. (2017). A convolutional neural network based feature learning and fault diagnosis method for the condition monitoring of gearbox. *Measurement* 111, 1–10. doi:10.1016/j.measurement.2017.07.017

Jolliffe, I. T. (1986). "Principal components in regression analysis," in *Principal component analysis* (New York, NY: Springer), 129–155.

Julier, S. J., and Uhlmann, J. K. (1997). New extension of the kalman filter to nonlinear systems. *Int. Symp. Aerospace/Defense Sensing, Simul. and Controls* 3068, 182–193.

Kadry, S. (2012). *Diagnostics and prognostics of engineering systems: methods and techniques: methods and techniques*. Hershey, PA: IGI Global.Google Scholar

Kennedy, J., and Eberhart, R. C. (1997). "A discrete binary version of the particle swarm algorithm," in 1997 IEEE International conference on systems, man, and cybernetics. Computational cybernetics and simulation (IEEE), Orlando, FL, 12–15, 1997, vol. 5, 4104–4108.

Khan, S., and Yairi, T. (2018). A review on the application of deep learning in system health management. *Mech. Syst. Signal Process.* 107, 241–265. doi:10.1016/j.ymssp.2017.11.024

Khelif, R., Chebel-Morello, B., Malinowski, S., Laajili, E., Fnaiech, F., and Zerhouni, N. (2017). Direct remaining useful life estimation based on support vector regression. *IEEE Trans. Ind. Electron.* 64, 2276–2285. doi:10.1109/tie.2016.2623260

Kingma, D. P., and Welling, M. (2013). Auto-encoding variational bayes. *arXiv*.

Kopparapu, S. K., and Laxminarayana, M. (2010). "Choice of mel filter bank in computing mfcc of a resampled speech," in 10th international conference on information science, signal processing and their applications (ISSPA 2010), Kuala Lumpur, Malaysia, May 2010 (IEEE, 121–124.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. *Commun. ACM* 60, 84. doi:10.1145/3065386

Kuo, R. J. (1995). Intelligent diagnosis for turbine blade faults using artificial neural networks and fuzzy logic. *Eng. Appl. Artif. Intell.* 8, 25–34. doi:10.1016/0952-1976(94)00082-x

Lal Senanayaka, J. S., Van Khang, H., and Robbersmyr, K. G. (2018). "Autoencoders and recurrent neural networks based algorithm for prognosis of bearing life," in 2018 21st International conference on electrical machines and systems (ICEMS), Jeju, South Korea, 537–542.

Lee, J., Wu, F., Zhao, W., Ghaffari, M., Liao, L., and Siegel, D. (2014). Prognostics and health management design for rotary machinery systems-Reviews, methodology and applications. *Mech. Syst. Signal Process.* 42, 314–334. doi:10.1016/j.ymssp.2013.06.004

Lei, J., Liu, C., and Jiang, D. (2019). Fault diagnosis of wind turbine based on long short-term memory networks. *Renew. Energy* 133, 422–432. doi:10.1016/j.renene.2018.10.031

Lei, Y., He, Z., and Zi, Y. (2009). A combination of wknn to fault diagnosis of rolling element bearings. *J. Vib. Acoust.* 131, 064502. doi:10.1115/1.4000478

Lei, Y., Yang, B., Jiang, X., Jia, F., Li, N., and Nandi, A. K. (2020). Applications of machine learning to machine fault diagnosis: a review and roadmap. *Mech. Syst. Signal Process.* 138, 106587. doi:10.1016/j.ymssp.2019.106587

Lei, Y., and Zuo, M. J. (2009). Gear crack level identification based on weighted k nearest neighbor classification algorithm. *Mech. Syst. Signal Process.* 23, 1535–1547. doi:10.1016/j.ymssp.2009.01.009

Li, C., Zhang, S., Qin, Y., and Estupinan, E. (2020). A systematic review of deep transfer learning for machinery fault diagnosis. *Neurocomputing* 407, 121–135. doi:10.1016/j.neucom.2020.04.045

Li, G., Deng, C., Wu, J., Xu, X., Shao, X., and Wang, Y. (2019a). Sensor data-driven bearing fault diagnosis based on deep convolutional neural networks and s-transform. *Sensors* 19, 2750. doi:10.3390/s19122750

Li, J., Li, X., and He, D. (2019b). A directed acyclic graph network combined with cnn and lstm for remaining useful life prediction. *IEEE Access* 7, 75464–75475. doi:10.1109/access.2019.2919566Google Scholar

Li, K. J., and Wang, Q. (2015). "Study on signal recognition and diagnosis for spacecraft based on deep learning method," in 2015 Prognostics and System Health Management Conference (PHM), Beijing, China, 1–5.

Li, X., Ding, Q., and Sun, J.-Q. (2018a). Remaining useful life estimation in prognostics using deep convolution neural networks. *Reliab. Eng. Syst. Saf.* 172, 1–11. doi:10.1016/j.ress.2017.11.021

Li, X., Jiang, H., Hu, Y., and Xiong, X. (2018b). "Intelligent fault diagnosis of rotating machinery based on deep recurrent neural network," in 2018 International conference on Sensing,Diagnostics, prognostics, and control (SDPC). Xi'an, China, 67–72.

Li, X., Zhang, W., and Ding, Q. (2019c). Deep learning-based remaining useful life estimation of bearings using multi-scale feature extraction. *Reliab. Eng. Syst. Saf.* 182, 208–218. doi:10.1016/j.ress.2018.11.011

Li, Z., Li, J., Wang, Y., and Wang, K. (2019d). A deep learning approach for anomaly detection based on sae and lstm in mechanical equipment. *Int. J. Adv. Manuf. Technol.* 103, 499. doi:10.1007/s00170-019-03557-w

Li, Z., Yan, X., Yuan, C., and Peng, Z. (2012). Intelligent fault diagnosis method for marine diesel engines using instantaneous angular speed. *J. Mech. Sci. Technol.* 26, 2413–2423. doi:10.1007/s12206-012-0621-2

Lipton, Z. C. (2018). The mythos of model interpretability. *Queue* 16, 31–57. doi:10.1145/3236386.3241340

Liu, H., Li, L., and Ma, J. (2016). Rolling bearing fault diagnosis based on stft-deep learning and sound signals. *Shock Vib.*, 2016, 1. doi:10.1155/2016/6127479

Liu, H., Zhou, J., Zheng, Y., Jiang, W., and Zhang, Y. (2018). Fault diagnosis of rolling bearings with recurrent neural network-based autoencoders. *ISA Transactions* 77, 167–178. doi:10.1016/j.isatra.2018.04.005

Liu, J., Li, Q., Chen, W., Yan, Y., Qiu, Y., and Cao, T. (2019a). Remaining useful life prediction of pemfc based on long short-term memory recurrent neural networks. *Int. J. Hydrogen Energy* 44, 5470–5480. doi:10.1016/j.ijhydene.2018.10.042.

Liu, X., Zhou, Q., Zhao, J., Shen, H., and Xiong, X. (2019b). Fault diagnosis of rotating machinery under noisy environment conditions based on a 1-d convolutional autoencoder and 1-d convolutional neural network. *Sensors* 19, 972. doi:10.3390/s19040972

Liu, Z., Zuo, M. J., and Xu, H. (2013). Feature ranking for support vector machine classification and its application to machinery fault diagnosis. *Proc. IME C J. Mech. Eng. Sci.* 227, 2077–2089. doi:10.1177/0954406212469757

Locatello, F., Bauer, S., Lucic, M., Raetsch, G., Gelly, S., Scho¨lkopf, B., et al. (2019a). "Challenging common assumptions in the unsupervised learning of disentangled representations," in Proceedings of the 36th international conference on machine learning. 4114–4124.

Locatello, F., Tschannen, M., Bauer, S., Rätsch, G., Schölkopf, B., and Bachem, O. (2019b). Disentangling factors of variation using few labels. *arXiv*.

Logan, D., and Mathew, J. (1996). Using the correlation dimension for vibration fault diagnosis of rolling element bearings-i. Basic concepts. *Mech. Syst. Signal Process.* 10, 241–250. doi:10.1006/mssp.1996.0018

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* 60, 91–110. doi:10.1023/B:VISI.0000029664.99615.94

Lu, C., Wang, Z.-Y., Qin, W.-L., and Ma, J. (2017). Fault diagnosis of rotary machinery components using a stacked denoising autoencoder-based health state identification. *Signal Process.* 130, 377. doi:10.1016/j.sigpro.2016.07.028

Lu, P.-J., Zhang, M.-C., Hsu, T.-C., and Zhang, J. (2001). An evaluation of engine faults diagnostics using artificial neural networks. *J. Eng. Gas Turbines Power* 123, 340. doi:10.1115/1.1362667

Lundberg, S. M., and Lee, S.-I. (2017). "A unified approach to interpreting model predictions," in *Advances in neural information processing systems 30*. Editors I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (New York, NY: Curran Associates, Inc.), 4765–4774.

Lv, F., Wen, C., Liu, M., and Bao, Z. (2017). Weighted time series fault diagnosis based on a stacked sparse autoencoder. *J. Chemometr.* 31, e2912. doi:10.1002/cem.2912. E2912 CEM-16-0169.R1

Ma, J., Su, H., Zhao, W.-l., and Liu, B. (2018). Predicting the remaining useful life of an aircraft engine using a stacked sparse autoencoder with multilayer self-learning. *Complexity*, 2018, 1–13. doi:10.1155/2018/3813029

Maaten, L. v. d., and Hinton, G. (2008). Visualizing data using t-sne. *J. Mach. Learn. Res.* 9, 2579–2605

Mao, W., He, J., Tang, J., and Li, Y. (2018). Predicting remaining useful life of rolling bearings based on deep feature representation and long short-term memory neural network. *Adv. Mech. Eng.* 10, 168781401881718. doi:10.1177/1687814018817184

Mao, W., Liu, Y., Ding, L., and Li, Y. (2019). Imbalanced fault diagnosis of rolling bearing based on generative adversarial network: a comparative study. *IEEE Access* 7, 9515–9530. doi:10.1109/access.2018.2890693

Mathew, V., Toby, T., Singh, V., Rao, B. M., and Kumar, M. G. (2017). "Prediction of remaining useful lifetime (rul) of turbofan engine using machine learning," in 2017 IEEE international conference on circuits and systems (ICCS), Thiruvananthapuram, 306–311.

McLachlan, G. J. (2004). *Discriminant analysis and statistical pattern recognition*, Vol. 544. Hoboken, NJ: John Wiley & Sons.

Mechefske, C. K., and Mathew, J. (1992). Fault detection and diagnosis in low speed rolling element bearings part ii: the use of nearest neighbor classification. *Mech. Syst. Signal Process.* 6, 309–316. doi:10.1016/0888-3270(92)90033-f

Medjaher, K., Tobon-Mejia, D. A., and Zerhouni, N. (2012). Remaining useful life estimation of critical components with application to bearings. *IEEE Trans. Reliab.* 61, 292–302. doi:10.1109/tr.2012.2194175

Michau, G., Hu, Y., Palmé, T., and Fink, O. (2019). Feature learning for fault detection in high-dimensional condition monitoring signals. *Proc. Inst. Mech. Eng. O J. Risk Reliab.* 234, 104–115. doi:10.1177/1748006x19868335

Michau, G., Palmé, T., and Fink, O. (2018). "Fleet phm for critical systems: bi-level deep learning approach for fault detection," in Proceedings of the Fourth European Conference of the Prognostics and Health Management Society, Utrecht, Netherlands, 4–6 July 2018, vol. 4.

Michau, G., Thomas, P., and Fink, O. (2017). "Deep feature learning network for fault detection and isolation," in PHM 2017: proceedings of the annual conference of the prognostics and health management society 2017, St. Petersburg, FL, 2-5 October 2017, 108–118.

Mitchell, T. M. (1997). *Machine learning*. 1st Edn. New York, NY: McGraw-Hill, Inc.

Mobley, R. K. (2002). *An introduction to predictive maintenance*. New York, NY: Elsevier.

Moosavi, S. S., N'Diaye, A., Djerdir, A., Ait-Amirat, Y., and Arab Khaburi, D. (2016). Artificial neural network-based fault diagnosis in the AC-DC converter of the power supply of series hybrid electric vehicle. *IET Electr. Syst. Transp.* 6, 96–106. doi:10.1049/iet-est.2014.0055

Moosavian, A., Ahmadi, H., Tabatabaeefar, A., and Khazaee, M. (2013). Comparison of two classifiers; k-nearest neighbor and artificial neural network, for fault diagnosis on a main engine journal-bearing. *Shock Vib.* 20, 263–272. doi:10.1155/2013/360236

Nascimento, G. R., and Viana, F. A. (2019). "Fleet prognosis with physics-informed recurrent neural networks," The 12th International Workshop on Structural

Health Monitoring 2019, Stanford, CA, September 10-12, 2019. doi:10.12783/shm2019/32301

Neath, A. A., and Cavanaugh, J. E. (2012). The bayesian information criterion: background, derivation, and applications. *WIREs Comp. Stat.* 4, 199–203. doi:10.1002/wics.199

Ng, A. (2011). Sparse autoencoder. *CS294A Lecture notes* 72, 1–19.

Ngui, W. K., Leong, M. S., Shapiai, M. I., and Lim, M. H. (2017). Blade fault diagnosis using artificial neural network. *Int. J. Appl. Eng. Res.* 12, 519–526.

Nik Aznan, N. K., Atapour-Abarghouei, A., Bonner, S., Connolly, J. D., Al Moubayed, N., and Breckon, T. P. (2019). "Simulating brain signals: creating synthetic eeg data via neural-based generative models for improved ssvep classification," in 2019 International joint conference on neural networks (IJCNN), Budapest, Hungary, 1–8.

Ordóñez, C., Sánchez Lasheras, F., Roca-Pardiñas, J., and Juez, F. J. d. C. (2019). A hybrid ARIMA-SVM model for the study of the remaining useful life of aircraft engines. *J. Comput. Appl. Math.* 346, 184–191. doi:10.1016/j.cam.2018.07.008

Pan, H., He, X., Tang, S., and Meng, F. (2018). An improved bearing fault diagnosis method using one-dimensional cnn and lstm. *J. Mech. Eng.* 64, 443–452.

Park, J., and Park, J. (2019). Physics-induced graph neural network: an application to wind-farm power estimation. *Energy* 187, 115883. doi:10.1016/j.energy.2019.115883

Park, P., Marco, P. D., Shin, H., and Bang, J. (2019). Fault detection and diagnosis using combined autoencoder and long short-term memory network. *Sensors* 19, 4612. doi:10.3390/s19214612

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., et al. (2019). "Pytorch: an imperative style, high-performance deep learning library," in *Advances in neural information processing systems*. Editors H. Wallach, H. Larochelle, A. Beygelzimer, E. Fox, and R. Garnett (Red Hook, NY: Curran Associates, Inc.), Vol. 32, 8024–8035.

Patil, S., Patil, A., Handikherkar, V., Desai, S., Phalle, V. M., and Kazi, F. S. (2018). "Remaining useful life (rul) prediction of rolling element bearing using random forest and gradient boosting technique," in *ASME international mechanical engineering congress and exposition* (New York, NY: American Society of Mechanical Engineers (ASME), Vol. 52187, V013T05A019.

Pearl, J. (2018). "Theoretical impediments to machine learning with seven sparks from the causal revolution," in Proceedings of the eleventh ACM international conference on web search and data mining, Marina Del Rey, CA, February 5–9, 2018

Poyhonen, S., Jover, P., and Hyotyniemi, H. (2004). "Signal processing of vibrations for condition monitoring of an induction motor," in First international symposium on control, communications and signal processing, 2004, Hammamet, Tunisia, 21–24 March 2004 (IEEE, 499–502.

Praveenkumar, T., Sabhrish, B., Saimurugan, M., and Ramachandran, K. I. (2018). Pattern recognition based on-line vibration monitoring system for fault diagnosis of automobile gearbox. *Measurement* 114, 233–242. doi:10.1016/j.measurement.2017.09.041

Qin, H., Xu, K., and Ren, L. (2019). "Rolling bearings fault diagnosis via 1d convolution networks," in 2019 IEEE 4th international Conference on Signal and image processing (ICSIP), Wuxi, China, July 19–21, 2019, 617–621.

Qiu, D., Liu, Z., Zhou, Y., and Shi, J. (2019). "Modified bi-directional lstm neural networks for rolling bearing fault diagnosis," in ICC 2019 - 2019 IEEE international conference on communications (ICC), Shanghai, China, May 20–24, 2019, 1–6.

Quinlan, J. R. (2014). *C4. 5: programs for machine learning*. New York, NY: Elsevier.

Ran, Y., Zhou, X., Lin, P., Wen, Y., and Deng, R. (2019). A survey of predictive maintenance: systems, purposes and approaches. *ArXiv*.

Rasmussen, C. E. (2003). "Gaussian processes in machine learning," in *Summer school on machine learning*. (New York, NY: Springer), 63–71.

Ren, L., Sun, Y., Cui, J., and Zhang, L. (2018). Bearing remaining useful life prediction based on deep autoencoder and deep neural networks. *J. Manuf. Syst.* 48, 71–77. doi:10.1016/j.jmsy.2018.04.008

Ren, L., Sun, Y., Wang, H., and Zhang, L. (2018a). Prediction of bearing remaining useful life with deep convolution neural network. *IEEE Access* 6, 13041–13049. doi:10.1109/access.2018.2804930

Ren, L., Zhao, L., Hong, S., Zhao, S., Wang, H., and Zhang, L. (2018b). Remaining useful life prediction for lithium-ion battery: a deep learning approach. *IEEE Access* 6, 50587–50598. doi:10.1109/access.2018.2858856

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). ""Why should i trust you?" explaining the predictions of any classifier," in Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. (New York, NY: ACM), 1135–1144.

Sakamoto, Y., Ishiguro, M., and Kitagawa, G. (1986). *Akaike information criterion statistics*. Dordrecht, The Netherlands: D. Reidel, Vol. 81.

Sakthivel, N. R., Sugumaran, V., and Babudevasenapati, S. (2010). Vibration based fault diagnosis of monoblock centrifugal pump using decision tree. *Expert Syst. Appl.* 37, 4040–4049. doi:10.1016/j.eswa.2009.10.002

Samanta, B. (2004). Gear fault detection using artificial neural networks and support vector machines with genetic algorithms. *Mech. Syst. Signal Process.* 18, 625–644. doi:10.1016/s0888-3270(03)00020-7

Samanta, B., and Al-Balushi, K. R. (2003). Artificial neural network based fault diagnostics of rolling element bearings using time-domain features. *Mech. Syst. Signal Process.* 17, 317–328. doi:10.1006/mssp.2001.1462

Sanchez-Gonzalez, A., Heess, N., Springenberg, J. T., Merel, J., Riedmiller, M., Hadsell, R., et al. (2018). Graph networks as learnable physics engines for inference and control. *arXiv*.

Santos, P., Villa, L., Reñones, A., Bustillo, A., and Maudes, J. (2015). An svm-based solution for fault detection in wind turbines. *Sensors* 15, 5627–5648. doi:10.3390/s150305627

Saravanan, N., and Ramachandran, K. I. (2009). Fault diagnosis of spur bevel gear box using discrete wavelet features and decision tree classification. *Expert Syst. Appl.* 36, 9564–9573. doi:10.1016/j.eswa.2008.07.089

Satishkumar, R., and Sugumaran, V. (2015). Remaining life time prediction of bearings through classification using decision tree algorithm. *Int. J. Appl. Eng. Res.* 10, 34861–34866.

Saxena, A., and Goebel, K. (2008). C-mapss data set. NASA ames prognostics data repository.

Schuster, M., and Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Trans. Signal Process.* 45, 2673–2681. doi:10.1109/78.650093

Shao, S., McAleer, S., Yan, R., and Baldi, P. (2019a). Highly accurate machine fault diagnosis using deep transfer learning. *IEEE Trans. Ind. Inf.* 15(4), 2446–2455. doi:10.1109/tii.2018.2864759

Shao, S., Wang, P., and Yan, R. (2019b). Generative adversarial networks for data augmentation in machine fault diagnosis. *Comput. Ind.* 106, 85–93. doi:10.1016/j.compind.2019.01.001

Shao, Y., and Nezu, K. (2000). Prognosis of remaining bearing life using neural networks. *Proc. IME J. Syst. Contr. Eng.* 214, 217–230. doi:10.1243/0959651001540582

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* 15, 1929–1958. doi:10.5555/2627435.2670313

Sugumaran, V. (2012). Exploiting sound signals for fault diagnosis of bearings using decision tree. *Measurement* 46, 1250–1256. doi:10.1016/j.measurement.2012.11.011.

Sugumaran, V., and Ramachandran, K. I. (2007). Automatic rule learning using decision tree for fuzzy classifier in fault diagnosis of roller bearing. *Mech. Syst. Signal Process.* 21, 2237–2247. doi:10.1016/j.ymssp.2006.09.007

Sui, W., Zhang, D., Qiu, X., Zhang, W., and Yuan, L. (2019). Prediction of bearing remaining useful life based on mutual information and support vector regression model. *IOP Conf. Ser. Mater. Sci. Eng.* 533, 012032. doi:10.1088/1757-899x/533/1/012032

Sun, C., Zhang, Z., and He, Z. (2011). Research on bearing life prediction based on support vector machine and its application. *J. Phys.: Conf. Ser.* 305, 012028. doi:10.1088/1742-6596/305/1/012028

Sun, K., Li, G., Chen, H., Liu, J., Li, J., and Hu, W. (2016a). A novel efficient SVM-based fault diagnosis method for multi-split air conditioning system's refrigerant charge fault amount. *Appl. Therm. Eng.* 108, 989. doi:10.1016/j.applthermaleng.2016.07.109

Sun, W., Shao, S., Zhao, R., Yan, R., Zhang, X., and Chen, X. (2016b). A sparse auto-encoder-based deep neural network approach for induction motor faults classification. *Measurement* 89, 171–178. doi:10.1016/j.measurement.2016.04.007

Sun, W., Yao, B., Zeng, N., Chen, B., He, Y., Cao, X., et al. (2017). An intelligent gear fault diagnosis methodology using a complex wavelet enhanced convolutional neural network. *Materials* 10, 790. doi:10.3390/ma10070790

Swanson, L. (2001). Linking maintenance strategies to performance. *Int. J. Prod. Econ.* 70, 237–244. doi:10.1016/s0925-5273(00)00067-0

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., et al. (2015). "Going deeper with convolutions," in Proceedings of the IEEE conference on computer vision and pattern recognition, Boston, MA, June 7–12, 2015, 1–9.

Tayade, A., Patil, S., Phalle, V., Kazi, F., and Powar, S. (2019). Remaining useful life (rul) prediction of bearing by using regression model and principal component analysis (pca) technique. *Vibroengineering PROCEDIA* 23, 30–36. doi:10.21595/vp.2019.20617

Teng, W., Zhang, X., Liu, Y., Kusiak, A., and Ma, Z. 2016). Prognosis of the remaining useful life of bearings in a wind turbine gearbox. *Energies* 10, 32. doi:10.3390/en10010032

Theano Development Team (2016). Theano: a python framework for fast computation of mathematical expressions. *arXiv*.

Thirukovalluru, R., Dixit, S., Sevakula, R. K., Verma, N. K., and Salour, A. (2016). "Generating feature sets for fault diagnosis using denoising stacked auto-encoder," in 2016 IEEE international conference on prognostics and health management (ICPHM), Ottawa, ON, Canada, June 20–22, 2016, 1–7.

Tian, J., Morillo, C., Azarian, M. H., and Pecht, M. (2016). Motor bearing fault detection using spectral kurtosis-based feature extraction coupled with k-nearest neighbor distance analysis. *IEEE Trans. Ind. Electron.* 63, 1793–1803. doi:10.1109/tie.2015.2509913

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *J. Roy. Stat. Soc. B* 58, 267–288. doi:10.1111/j.2517-6161.1996.tb02080.x

Tzeng, E., Hoffman, J., Saenko, K., and Darrell, T. (2017). "Adversarial discriminative domain adaptation," in Proceedings of the IEEE conference on computer vision and pattern recognition, Honolulu, HI, July 21–26, 2017, 7167–7176.

Vincent, P., Larochelle, H., Bengio, Y., and Manzagol, P.-A. (2008). "Extracting and composing robust features with denoising autoencoders," in Machine learning, proceedings of the twenty-fifth international conference (ICML 2008), Helsinki, Finland, June 5–9, 2008, 1096–1103. doi:10.1145/1390156.1390294

Wang, J., Li, S., Han, B., An, Z., Bao, H., and Ji, S. (2019). Generalization of deep neural networks for imbalanced fault classification of machinery using generative adversarial networks. *Ieee Access* 7, 111168–111180. doi:10.1109/access.2019.2924003

Wang, J., Mo, Z., Zhang, H., and Miao, Q. (2019). A deep learning method for bearing fault diagnosis based on time-frequency image. *IEEE Access* 7, 42373–42383. doi:10.1109/access.2019.2907131

Wang, L., Zhao, X., Pei, J., and Tang, G. (2016). Transformer fault diagnosis using continuous sparse autoencoder. *SpringerPlus* 5, 448. doi:10.1186/s40064-016-2107-7

Wang, Q., Michau, G., and Fink, O. (2019a). "Domain adaptive transfer learning for fault diagnosis," in 2019 prognostics and system health management conference, Paris, France, May 2–5, 2019 (IEEE), 279–285.

Wang, Q., Zhao, B., Ma, H., Chang, J., and Mao, G. (2019b). A method for rapidly evaluating reliability and predicting remaining useful life using two-dimensional convolutional neural network with signal conversion. *J. Mech. Sci. Technol.* 33, 2561–2571. doi:10.1007/s12206-019-0504-x

Wang, X., and Liu, F. (2020). Triplet loss guided adversarial domain adaptation for bearing fault diagnosis. *Sensors* 20, 320. doi:10.3390/s20010320

Wang, X.-X., and Ma, L.-Y. (2014). A compact k nearest neighbor classification for power plant fault diagnosis. *J. Inf. Hiding Multimed. Signal Process* 5, 508–517.

Wang, Z., Zhang, Q., Xiong, J., Xiao, M., Sun, G., and He, J. (2017). Fault diagnosis of a rolling bearing using wavelet packet denoising and random forests. *IEEE Sensor. J.* 17, 5581–5588. doi:10.1109/jsen.2017.2726011

Wei, J., Dong, G., and Chen, Z. (2018). Remaining useful life prediction and state of health diagnosis for lithium-ion batteries using particle filter and support vector regression. *IEEE Trans. Ind. Electron.* 65, 5634–5643. doi:10.1109/tie.2017.2782224

Wen, J., and Gao, H. (2018). Degradation assessment for the ball screw with variational autoencoder and kernel density estimation. *Adv. Mech. Eng.* 10, 168781401879726. doi:10.1177/1687814018797261

Wen, L., Dong, Y., Dong, Y., and Gao, L. 2019a). A new ensemble residual convolutional neural network for remaining useful life estimation. *Math. Biosci. Eng.* 16, 862–880. doi:10.3934/mbe.2019040

Wen, L., Li, X., and Gao, L. (2019b). A transfer convolutional neural network for fault diagnosis based on resnet-50. *Neural Comput. Appl.* 32, 6111–6124. doi:10.1007/s00521-019-04097-w

Wen, L., Li, X., Gao, L., and Zhang, Y. (2018). A new convolutional neural network-based data-driven fault diagnosis method. *IEEE Trans. Ind. Electron.* 65, 5990–5998. doi:10.1109/tie.2017.2774777

Widodo, A., and Yang, B.-S. (2007). Application of nonlinear feature extraction and support vector machines for fault diagnosis of induction motors. *Expert Syst. Appl.* 33, 241–250. doi:10.1016/j.eswa.2006.04.020

Wu, J., Hu, K., Cheng, Y., Zhu, H., Shao, X., and Wang, Y. (2020). Data-driven remaining useful life prediction via multiple sensor signals and deep long short-term memory neural network. *ISA (Instrum. Soc. Am.) Trans.* 97, 241–250. doi:10.1016/j.isatra.2019.07.004

Wu, Q., Ding, K., and Huang, B. (2018a). Approach for fault prognosis using recurrent neural network. *J. Intell. Manuf.*, 31, 1621. doi:10.1007/s10845-018-1428-5

Wu, Y., Yuan, M., Dong, S., Lin, L., and Liu, Y. (2018b). Remaining useful life estimation of engineered systems using vanilla lstm neural networks. *Neurocomputing* 275, 167–179. doi:10.1016/j.neucom.2017.05.063

Xia, M., Li, T., Shu, T., Wan, J., de Silva, C. W., and Wang, Z. (2019). A two-stage approach for the remaining useful life prediction of bearings using deep neural networks. *IEEE Trans. Ind. Inf.* 15, 3703–3711. doi:10.1109/tii.2018.2868687

Xiang, S., Qin, Y., Zhu, C., Wang, Y., and Chen, H. (2020). Long short-term memory neural network with weight amplification and its application into gear remaining useful life prediction. *Eng. Appl. Artif. Intell.* 91, 103587. doi:10.1016/j.engappai.2020.103587

Xueyi, L., Li, J., Qu, A., and He, D. (2019). Gear pitting fault diagnosis using integrated cnn and gru network with both vibration and acoustic emission signals. *Appl. Sci.* 9, 768. doi:10.3390/app9040768

Yan, H., Wan, J., Zhang, C., Tang, S., Hua, Q., and Wang, Z. (2018). Industrial big data analytics for prediction of remaining useful life based on deep learning. *IEEE Access* 6, 17190–17197. doi:10.1109/access.2018.2809681

Yan, W. (2006). "Application of random forest to aircraft engine fault diagnosis," in The proceedings of the multiconference on" computational engineering in systems applications", Beijing, China, October 4–6, 2006 (IEEE), Vol. 1, 468–475

Yan, W. (2019). Detecting gas turbine combustor anomalies using semi-supervised anomaly detection with deep representation learning. *Cogn Comput* 12, 1–14. doi:10.1007/s12559-019-09710-7

Yang, B., Liu, R., and Zio, E. (2019). Remaining useful life prediction based on a double-convolutional neural network architecture. *IEEE Trans. Ind. Electron.* 66, 9521–9530. doi:10.1109/tie.2019.2924605

Yang, B.-S., Di, X., and Han, T. (2008). Random forests classifier for machine fault diagnosis. *J. Mech. Sci. Technol.* 22, 1716–1725. doi:10.1007/s12206-008-0603-6

Yang, B.-S., Han, T., and Hwang, W.-W. (2005). Fault diagnosis of rotating machinery based on multi-class support vector machines. *J. Mech. Sci. Technol.* 19, 846–859. doi:10.1007/BF02916133

Yang, J., Zhang, Y., and Zhu, Y. (2007). Intelligent fault diagnosis of rolling element bearing based on svms and fractal dimension. *Mech. Syst. Signal Process.* 21, 2012–2024. doi:10.1016/j.ymssp.2006.10.005

Yang, Z.-X., Wang, X.-B., and Zhong, J.-H. (2016). Representational learning for fault diagnosis of wind turbine equipment: a multi-layered extreme learning machines approach. *Energies* 9, 379. doi:10.3390/en9060379

Yao, Y., Wang, H., Li, S., Liu, Z., Gui, G., Dan, Y., et al. (2018). End-to-end convolutional neural network model for gear fault diagnosis based on sound signals. *Appl. Sci.* 8, 1584. doi:10.3390/app8091584

Yoo, Y., and Baek, J.-G. (2018). A novel image feature for the remaining useful lifetime prediction of bearings based on continuous wavelet transform and convolutional neural network. *Appl. Sci.* 8, 1102. doi:10.3390/app8071102

Yoon, J., Jarrett, D., and van der Schaar, M. (2019). "Time-series generative adversarial networks," in Advances in neural information processing systems Editors H. Wallach, H. Larochelle, A. Beygelzimer, F. d' Alché-Buc, E. Fox, and R. Garnett (New York, Ny: Curran Associates, Inc.), 5508–5518.

Yosinski, J., Clune, J., Bengio, Y., and Lipson, H. (2014). "How transferable are features in deep neural networks?," in *Advances in neural information processing systems 27*. Editors. Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger (New York, NY: Curran Associates, Inc.), 3320–3328.

Yuan, J., and Tian, Y. (2019). An intelligent fault diagnosis method using gru neural network toward sequential data in dynamic processes. *Processes* 7, 152. doi:10.3390/pr7030152

Yuan, M., Wu, Y., and Lin, L. (2016). "Fault diagnosis and remaining useful life estimation of aero engine using lstm neural network," in 2016 IEEE international conference on aircraft utility systems (AUS), Beijing, China, October 10–12, 2016, 135–140.

Yuan, Z., Zhang, L., and Duan, L. (2018). A novel fusion diagnosis method for rotor system fault based on deep learning and multi-sourced heterogeneous monitoring data. *Meas. Sci. Technol.* 29, 115005. doi:10.1088/1361-6501/aadfb3

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. (2017). "Understanding deep learning requires rethinking generalization," in 5th international conference on learning representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference track proceedings (OpenReview.net).

Zhang, Y., Xiong, R., He, H., and Pecht, M. G. (2018). Long short-term memory recurrent neural network for remaining useful life prediction of lithium-ion batteries. *IEEE Trans. Veh. Technol.* 67, 5695–5705. doi:10.1109/tvt.2018.2805189

Zhao, H., Sun, S., and Jin, B. (2018). Sequential fault diagnosis based on lstm neural network. *IEEE Access* 6, 12929–12939. doi:10.1109/access.2018.2794765

Zhao, Q., Qin, X., Zhao, H., and Feng, W. (2018). A novel prediction method based on the support vector regression for the remaining useful life of lithium-ion batteries. *Microelectron. Reliab.* 85, 99–108. doi:10.1016/j.microrel.2018.04.007

Zhao, R., Yan, R., Chen, Z., Mao, K., Wang, P., and Gao, R. X. (2019). Deep learning and its applications to machine health monitoring. *Mech. Syst. Signal Process.* 115, 213–237. doi:10.1016/j.ymssp.2018.05.050

Zhao, R., Yan, R., Wang, J., and Mao, K. (2017). Learning to monitor machine health with convolutional bi-directional lstm networks. *Sensors* 17, 273. doi:10.3390/s17020273

Zheng, H., Wang, R., Yang, Y., Yin, J., Li, Y., Li, Y., et al. (2019). Cross-domain fault diagnosis using knowledge transfer strategy: a review. *IEEE Access* 7, 129260–129290. doi:10.1109/access.2019.2939876

Zheng, S., Ristovski, K., Farahat, A., and Gupta, C. (2017). "Long short-term memory network for remaining useful life estimation," in 2017 IEEE international conference on prognostics and health management (ICPHM), Dallas, TX, June 19–21, 2017, 88–95

Zheng, Z., Peng, J., Deng, K., Gao, K., Li, H., Chen, B., et al. (2019). "A novel method for lithium-ion battery remaining useful life prediction using time window and gradient boosting decision trees," in 2019 10th international conference on power electronics and ECCE Asia (ICPE 2019 - ECCE Asia), Busan, South Korea, May 27–30, 2019, 3297–3302.

Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., and Torralba, A. (2016). "Learning deep features for discriminative localization," in Proceedings of the IEEE conference on computer vision and pattern recognition, Las Vegas, NV, 2921–2929.

Zhu, H., Ting, R., Wang, X., Zhou, Y., and Fang, H. (2015). "Fault diagnosis of hydraulic pump based on stacked autoencoders," in 2015 12th IEEE international conference on electronic measurement Instruments (ICEMI), Vol. 01, 58–62.

Zhu, J., Chen, N., and Peng, W. (2018). Estimation of bearing remaining useful life based on multiscale convolutional neural network. *IEEE Trans. Ind. Electron.* 66, 3208–3216.

Conflict of Interest: Authors IK and LB are employed by the company CSEM SA.

The authors declare that this study received funding from CSEM SA. The funder was not involved in the study design, collection, analysis, interpretation of data, the writing of this article or the decision to submit it for publication.

frontiers in Artificial Intelligence

ORIGINAL RESEARCH
published: 16 November 2020
doi: 10.3389/frai.2020.578152

Predictive Maintenance for Injection Molding Machines Enabled by Cognitive Analytics for Industry 4.0

Vaia Rousopoulou, Alexandros Nizamis, Thanasis Vafeiadis, Dimosthenis Ioannidis and Dimitrios Tzovaras*

Centre for Research and Technology Hellas-Information Technologies Institute (CERTH/ITI), Thessaloniki, Greece

The exploitation of big volumes of data in Industry 4.0 and the increasing development of cognitive systems strongly facilitate the realm of predictive maintenance for real-time decisions and early fault detection in manufacturing and production. Cognitive factories of Industry 4.0 aim to be flexible, adaptive, and reliable, in order to derive an efficient production scheme, handle unforeseen conditions, predict failures, and aid the decision makers. The nature of the data streams available in industrial sites and the lack of annotated reference data or expert labels create the challenge to design augmented and combined data analytics solutions. This paper introduces a cognitive analytics, self- and autonomous-learned system bearing predictive maintenance solutions for Industry 4.0. A complete methodology for real-time anomaly detection on industrial data and its application on injection molding machines are presented in this study. Ensemble prediction models are implemented on the top of supervised and unsupervised learners and build a compound prediction model of historical data utilizing different algorithms' outputs to a common consensus. The generated models are deployed on a real-time monitoring system, detecting faults in real-time incoming data streams. The key strength of the proposed system is the cognitive mechanism which encompasses a real-time self-retraining functionality based on a novel double-oriented evaluation objective, a data-driven and a model-based one. The presented application aims to support maintenance activities from injection molding machines' operators and demonstrate the advances that can be offered by exploiting artificial intelligence capabilities in Industry 4.0.

Keywords: cognitive analytics, artificial intelligence in manufacturing, predictive maintenance, ensemble learning, injection molding, Industry 4.0

OPEN ACCESS

Edited by:
Oscar Lazaro,
Innovalia Association, Spain

Reviewed by:
Arafat Salih Aydiner,
Istanbul Medeniyet University, Turkey
Saeed Tabar,
Ball State University, United States

****Correspondence:***
Vaia Rousopoulou
vrousop@iti.gr

Specialty section:
This article was submitted to
AI in Business,
a section of the journal
Frontiers in Artificial Intelligence

Received: *30 June 2020*
Accepted: *24 September 2020*
Published: *16 November 2020*

Citation:
Rousopoulou V, Nizamis A, Vafeiadis T, Ioannidis D and Tzovaras D (2020) Predictive Maintenance for Injection Molding Machines Enabled by Cognitive Analytics for Industry 4.0. Front. Artif. Intell. 3:578152. doi: 10.3389/frai.2020.578152

1 INTRODUCTION

Nowadays, the continuous accelerating pace of data creation and gathering from a wide range of sources such as sensors, posts to social media sites, transaction records, traffic data, pictures and videos, health data, mobile devices, and users' activities led to significant changes in data analytics solutions by boosting machine learning (ML) and artificial intelligence (AI) methodologies to a wide range of domains (Salamanis et al., 2016; Vatrapu et al., 2016; Galetsi et al., 2020). The manufacturing domain was not an exception. The adoption of state-of-the-art algorithms and cutting-edge technologies in the years of Industry 4.0 enables the automation of processes and the creation of novel predictive maintenance solutions based on predictive and prescriptive analytics (Rojko, 2017).

Nonetheless, the full potential of the fast growing and changing data in manufacturing domain has not been unlocked yet. The application of human-like intelligence in the form of cognitive analytics in manufacturing domain is still in initial stages. Some initial approaches for cognitive manufacturing manage to improve analytics services' quality and consistency. However, cognitive applications that can get smarter and more effective over time by learning from their interaction with data and by evaluating their own performance indicators in terms of precision, is still an ongoing activity. To this aim, the work presented in this study introduces a cognitive framework that exploits the capabilities of retraining mechanisms by continuous learning. Its application for predictive maintenance services in injection molding machines of a large electronics manufacturer's shop floor demonstrates the advantages of this cognitive solution in terms of predictions' accuracy.

An injection molding machine is commonly used in plastic processing industry and has to work continuously for long hours, so as to enable a continuous production line. A series of prediction, prevention, and inspection activities in order to alert machine problems and failures are vital for the normal and stable operation of a molding machine. This category of machines consists of different parts such as hydraulic, mechanical, and electrical parts that can cause failures. Usually, a failure is related to abnormal rise of temperature in an injection molding machine. The problem can be related to various factors such as problems in cooling system, improper pressure regulator, and high pressure in hydraulic system alongside with long period of overheating. Besides the temperature, the abnormal generated noise can be a real-time failure indicator if this kind of data is available. Damaged hydraulic and mechanical components can lead to significant variations of sound. The detection of substandard products with lower quality could be the last indicator of an injection molding machine failure.

The current study introduces a predictive solution based on the application of cognitive analytics in feature parameters coming in real time from injection molding machines by using IDS connectors (Otto et al., 2019; Otto and Jarke, 2019). The proposed predictive models aim to detect abnormalities out of the available temperature, pressure, and energy consumption data. Since both labeled and unlabeled data exist in the aforementioned machines, supervised and unsupervised learning algorithms have been deployed based on the availability or not of ground truth in the data, respectively. Ensemble learning was implemented upon different learners in order to combine their independent decisions and boost the fault detection mechanism. In the case that ground truth is available by the machines, the Adaptive Boosting (Freund and Schapire, 1995; Nath and Behara, 2003; Schapire and Freund, 2012) ensemble technique was applied to the deployed supervised learning methodologies in order to increase predictive performance. Accordingly, the major voting method is implemented on the top of unsupervised learning. As the injection molding machine condition monitoring forms a nonstationary environment, an adaptive and evolving approach is presented, capable of accommodating changes. So, the produced predictive models are continuously evaluated through a double-oriented evaluation objective, a data-driven and a model-based one. The latter enables a novel real-time self-retraining functionality for boosting the cognitive capabilities of the proposed solution.

The paper is structured as follows. Following the Introduction, a related work review is presented. **Section 3** contains a detailed description of the proposed methodology, while **Section 4** demonstrates the experimental results of the study. Finally, the conclusions of the study are drawn at **Section 5**.

2 RELATED WORK

There are several available methodologies, concepts, and solutions related to predictive maintenance services in Industry 4.0. The selected related work in this section is presented by the perspectives of cognition in manufacturing domain, predictive maintenance for injection molding machines, and ensemble methods for the enhancement of predictive services. The three aforementioned categories constitute the main advances of the current work and the corresponding bibliography was considered as the most suitable one to be mentioned in this section.

The advances in nowadays software and hardware technologies enable computer systems to mimic human brain activities and acquire cognition capabilities. The alleged capabilities introduce cognitive computing which is based on software that learns by itself, without reprogramming, and it is able to automate cognitive tasks. Industry 4.0 solutions have adopted various cognitive computing approaches for predictive maintenance, planning optimization, and performance and quality improvement. To this direction, the concept of the Cognitive Factory is supposed to be flexible, adaptable, reliable, and efficient in various momentary situations (Zaeh et al., 2009). This type of factory is moving from perception to action by using continuous learning and cognitive mechanisms. The advantages, disadvantages, and future challenges in the field of cognitive manufacturing have been widely studied (Bannat et al., 2011; Iarovyi et al., 2015). Iarovyi et al. (2015) present a documentation of different architectures for cognitive manufacturing systems that can be benefited from Industrial Internet of Things and cognitive control. Bannat et al. (2011) investigate methods to realize cognitive control and cognitive operation of production systems by highlighting self-optimizing and self-learning procedures. Iarovyi et al. (2015) again propose an architecture for cognitive manufacturing systems by combining approaches from PLANTCockpit (2012) and CogNetCon (Boza et al., 2011), enabling efficient data integration in manufacturing environments and providing connectivity between data on shop floor level and data in MES, ERP, and other systems. A cognition layer in the architecture contains a cognition engine, a model repository, and knowledge representation components. By adopting the aforementioned components, the architecture targets higher-level decision-making, self-learning, reconfiguration, and self-optimization in manufacturing domain.

Comprehensive research has been held toward predictive maintenance in manufacturing, including the study and analysis of sensor data and industrial machines for early fault detection, condition base monitoring, and decision support systems. Specifically, injection molding machines have been investigated as a real-world industrial application of predictive analytics (Gatica et al., 2016; Park et al., 2016; Jankov et al., 2017). An overview of industrial analytics methods and applications for predictive maintenance in manufacturing is presented by Gatica et al. (2016), encompassing an injection molding machines' use case. The work of Gatica et al. (2016) classifies the machinery analytic approaches in offline and online analysis. The offline analytics contain the "hypothesis-driven" strategy which is based on the analysis of the machine behavior and the "data-driven" strategy which focuses on exploration of the information provided by sensors and machine logs. Online analytics resolve predictive maintenance through data monitoring and machine state recognition by employing machine learning models. Jankov et al. (2017) introduce another real-time anomaly detection system dealing with injection molding machines. The presented system performs anomaly detection using K-means for cluster finding and Markov model for data training. Jankov et al. (2017) describe a custom-built system and concentrate on the system's performance through parallel and real-time processes. Furthermore, the method proposed by Park et al. (2016) distinguishes the different maintenance items of an injection molding and maps each one of these items to selected parameters in the collected data. Thereafter, a live parameters' monitoring process takes place and abnormal trends or patterns are detected based on statistical techniques. The detected abnormalities for different machine's parts are available to the maintenance operators. Last but not least, in the field of predictive analytics in manufacturing, a study which introduces an application on industrial ovens is worth mentioning (Rousopoulou et al., 2019). This study's methodology could as well be applied to injection molding machines, as it concerns the usage of both existing machine sensors with their log data and deployed sensors and achieves early fault diagnosis in an industrial machine.

Finally, on the subject of ensemble learning, ensemble techniques contribute to the performance of supervised and unsupervised machine learning models and enhance the predictive maintenance analytics solutions. A recent work regarding ensemble learning proves the improvement of individual learning models in terms of accuracy as well as training time by implementing ensemble learning and creating an integrated model through majority voting, experimenting on refrigerator system's datasets (Zhang et al., 2020). Additionally, in terms of assessing the ensemble techniques, a thorough benchmarking evaluation of outlier detection algorithms was reviewed (Domingues et al., 2018). Unsupervised machine learning algorithms were tested and compared on multiple datasets, highlighting their strengths and weaknesses. Within this context, an application of unsupervised outlier detection on streaming data containing travel booking information was implemented (Domingues et al., 2016). The study of Domingues et al. (2016) performs fraud detection by examining aggregation functions and interpolation in order to address unsupervised ensemble learning.

3 METHODOLOGY

The proposed methodology constructs a real-time anomaly detection solution with cognitive retraining, applied to an industrial machine. Starting with the training of historical data by state-of-the-art ML algorithms and meta-learners, prediction models are created. The models are fed with live incoming data streams and detect abnormalities in real time. This live monitoring process is enhanced by an automated retraining mechanism which inspects the characteristics of the new input data and the models' performance in order to update the prediction models and maintain the high performance of the fault detection system. The methodology consists of the following components:

(1) Data Preparation
(2) Online Training
(3) Ensemble Learning
(4) Live Prediction
(5) Cognitive Check

Figure 1 is an illustration of the proposed methodology. The process starts with historical data that are inserted for data preparation. The online training includes the training of several algorithms which are afterward enhanced by the ensemble learning step. The above steps examine and determine the optimum models for live prediction which performs real-time anomaly detection. New live data are coming through the predictive models which are constantly evaluated by a cognitive check and updated by automatic model retraining. The proposed method aims to form a regularly updated system which can monitor an injection molding machine and predict machine or part failures in order to reduce or even prevent machine downtime and save time and cost in the production line. The remainder of this chapter is a detailed description of the proposed solution, underlining the methods and algorithms combined in a complete anomaly detection pipeline.

3.1 Data Preparation

The current study's dataset is composed of measurements from injection molding machines which carry out the process of shaping rubber or plastic parts by injecting heated material into a mold. Specifically, the injection molding process deals with the fabrication of plastic components for electric shavers. The available measurements are expressed in time-series format, including different kinds of measurements, such as temperature, pressure, energy consumption, and time. Anomalies in time-series data indicate "bad" shots during injection, which leads to rejected products.

Six different injection molding machines are available in the dataset; four of them contain labeled data and the other two contain unlabeled data. **Table 1** shows the features that each

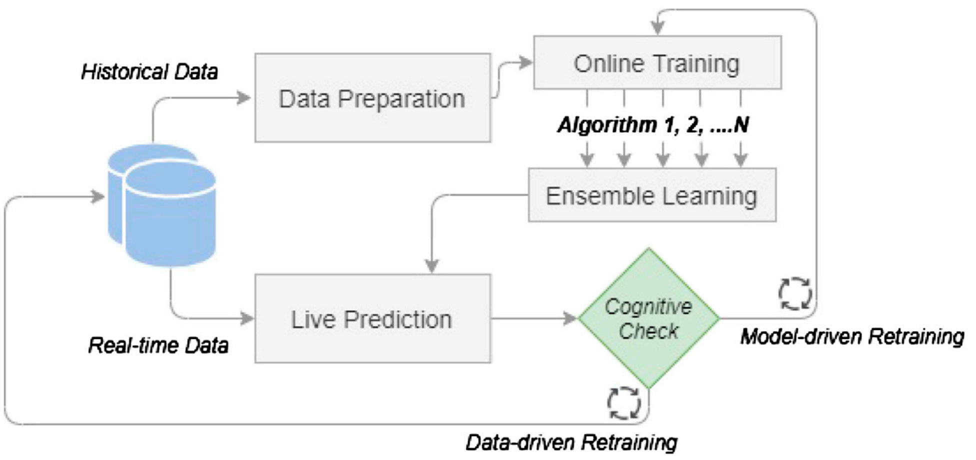


FIGURE 1 | Overview of the cognitive system's methodology architecture.

TABLE 1 | The available datasets from injection molding machines.

Feature description	Injection molding machine #
Timestamp	1, 2, 3, 4, 5, 6
Part counter	1, 2
Bad part counter indicator	1, 2
Last value of cycle time	1, 2, 3, 4, 5, 6
Peak of hold pressure	1, 2, 3, 4, 5, 6
Injection pressure	1, 2
Peak of injection pressure	1, 2, 5, 6
Flow number	1, 2, 3, 4, 5, 6
Melt cushion	1, 2, 3, 4, 5, 6
Hydraulic pressure	1, 2
Peak of injection pressure	1, 2, 3, 4, 5, 6
Value of switchover position	1, 2
Corrected position of plasticizing	1, 2, 3, 4, 5, 6
Clamp force	1, 2, 3, 4, 5, 6
Value of mold protection time	1, 2, 5, 6
Oil temperature	1, 2, 3, 4
Temperature of zone X	1, 2, 3, 4, 5, 6
Number of cavities	3, 4
Cooling time	3, 4, 5, 6
Screw position	3, 4, 5, 6
Injection time	3
Switchover pressure	3, 4, 5, 6
Shot counter	3, 4
Bad shot counter	3, 4
Peak of back pressure	3, 4, 5, 6
Plasticizing time	3, 4, 5, 6
Set value of temperature of zone X	3, 4, 5, 6
Heating energy consumption	3, 4, 5, 6
Motor energy consumption	3, 4, 5, 6
Total energy consumption	3, 4, 5, 6
Quality indicator (label)	3, 4, 5, 6

machine contains. The labeled datasets include a quality indicator feature (label). The zero label suggests a normal instance, while a nonzero label indicates an abnormality in machine's performance. The values of the quality indicator feature correspond to a specific error, but since this aspect goes beyond the context of the current research, the label is converted to binary, with zero meaning normal and one meaning abnormal instance.

TABLE 2 | The available datasets from injection molding machines.

Dataset	Label	Initial feature size	Final feature size
Injection molding machine 1	Yes	63	48
Injection molding machine 2	Yes	63	47
Injection molding machine 3	Yes	75	42
Injection molding machine 4	Yes	73	47
Injection molding machine 5	No	34	23
Injection molding machine 6	No	34	21

In order to transform raw data into refined information assets, first cleansing of the data takes place. The constant, empty, and duplicated columns are removed from the dataset. The columns with insignificant variance measure, namely, lower than 0.01, are removed as well. In order to resolve the real-world data inconsistency or incompleteness, the following preprocessing and cleansing methods are implemented for the data preparation step:

- Interpolation for estimating missing values between known data points is used.
- Zero variables are eliminated by zero removal process, as well.
- Normalization is used in order to scale and translate each feature individually in the range between zero and one.

After the preprocessing phase the features of each dataset are reduced as shown in **Table 2**. The dataset is split into 70% of samples for training and 30% for testing.

3.2 Online Training

A set of supervised and unsupervised learners is applied to the machine data, depending on the needs of each injection molding machine. The nature of the data led us to address the anomaly detection problem through classification and clustering methods. The algorithms reported at this chapter were selected for the current research due to the sufficient results that they have achieved in terms of prediction models' performance. However, the solution is extensible enough, so as to incorporate new methods within the overall architecture.

The online training of labeled datasets is addressed by well-known supervised training methods. The methods classify the input datasets and are capable of creating prediction models that detect faults in this input. Specifically, Support Vector Machines (SVMs) classifier is one of the most convenient and widespread classification algorithms, able to construct a hyperplane as a decision boundary as the maximum margin between classified classes based on kernel functions. In this work, two kernel functions are applied: Polynomial and Radial Basis Function. Decision tree learning is a technique for approximating discrete-valued functions, in which the learned function is represented by a decision tree (or classification tree or learning tree) (Lee and Siau, 2001). Random forest is an ensemble of decision trees and each decision tree is constructed by using a random subset of the training data, while the output class is the mode of the classes decided by each decision tree (Breiman, 1999). Finally, Artificial Neural Networks (ANNs) are used and especially Back Propagation Network (BPN) which is a feed-forward model with supervised learning (Rumelhart et al., 1986), and for the need of this work a fully connected neural network is used with one hidden layer.

On the other side, for the online training of unlabeled datasets, unsupervised learning techniques were implemented aiming to detect anomalies through data clustering. Thus, in this section we present state-of-the-art unsupervised learning methodologies that have been used in this work. DBSCAN is the data clustering algorithm which discovers clusters of arbitrary shape in spatial dataspaces with noise. Next is the Local Outlier Factor (LOF), which provides a factor of how close a data point is to its neighbors in respect to its neighbor being also close to it. The One-Class Support Vector Machine (One-Class SVM) algorithm classifies the points that lie outside some boundaries of the data space as outliers. Finally, K-means iteratively tries to partition the dataset into clusters with each data point belonging to only one cluster.

3.3 Ensemble Learning

On the top of individual learners, ensemble methods are techniques that utilize multiple models so as to combine them in order to produce improved results. Ensemble methods are incorporated into our methodology so as to generate a more accurate solution comparing with the results of single models. Our methodology proposes two ensemble algorithms for supervised and unsupervised learners, Adaptive Boosting and majority voting, respectively.

Adaptive Boosting (or AdaBoost) technique is a conjunction of many classification algorithms (also called weak learners), either from different families or from the same family with different internal parameters, aiming to improve classification performance compared to a single and simple classification algorithm. AdaBoost takes as input the outcome of a weak learner and iteratively improve it by recalculating its weights for the incorrectly classified cases in the training set. Adaboost is adaptive in the sense that subsequent weak learners are tweaked in favor of those instances misclassified by previous classifiers. There are many forms of boosting algorithms (Nath and Behara, 2003; Schapire and Freund, 2012), but the most popular is the one where the weak classifiers are decision trees (Freund and Schapire, 1995). In this work, we use the AdaBoost SAMME–Stagewise Additive Modeling using multiclass exponential loss function, which is an extension of AdaBoost.M1 algorithm, so as to perform both two-class and multiclass classification scenarios.

Majority vote (Jung and Lease, 2012) is a simple method for generating consensus among different algorithms by picking the label receiving the most votes. The rationale of the method is to calculate the average label coming from multiple learners and round according to a decision threshold. The majority vote is used as an enhancement for the individual learner's anomaly detection. It is also used as a replacement for the ground truth in case of unlabeled datasets, especially on the cognitive check taking place in the live prediction step of the proposed methodology (**Section 3.5**).

The aforementioned ensemble methods are applied automatically to the trained algorithms. In the case of supervised learners the AdaBoost methods provide an enhancement in terms of accuracy of single learners. In the case of unsupervised learners, the majority voting is used as a combined learner which performs better than a single learner or as a substitute of ground truth values in order to facilitate the cognitive check described in the next sections.

3.4 Live Prediction

When the training phase is over each machine dataset acquires one prediction model and its metadata (preprocessing models, statistic measures, and logs). The prediction model with the highest accuracy metric prevails in case of supervised learning, whereas the model with the highest silhouette score prevails in case of unsupervised models. Both supervised and unsupervised optimal models are specified as the "default" model. These models perform the outlier prediction on new live incoming data streams. The machine data are constantly monitored and for each instance of measurements, the system recognizes normal behavior or detects anomalies.

The incoming data stream is being edited and brought to the same format as the training dataset. The preprocessing methods used in training phase are applied precisely to the input data stream which will next be imported in the "default" prediction model. The anomaly detection results are kept in order to be used for evaluation and cognitive updating. The live monitoring procedure is constantly operating and updated in the aforementioned way.

Technically, the injection molding machine live data are retrieved through a custom IDS connector system that was set up for the purposes of the presented work. **Figure 2** illustrates the integration of real-time machine data with the cognitive analytics. The system is based on two IDS Trusted Connectors. The first

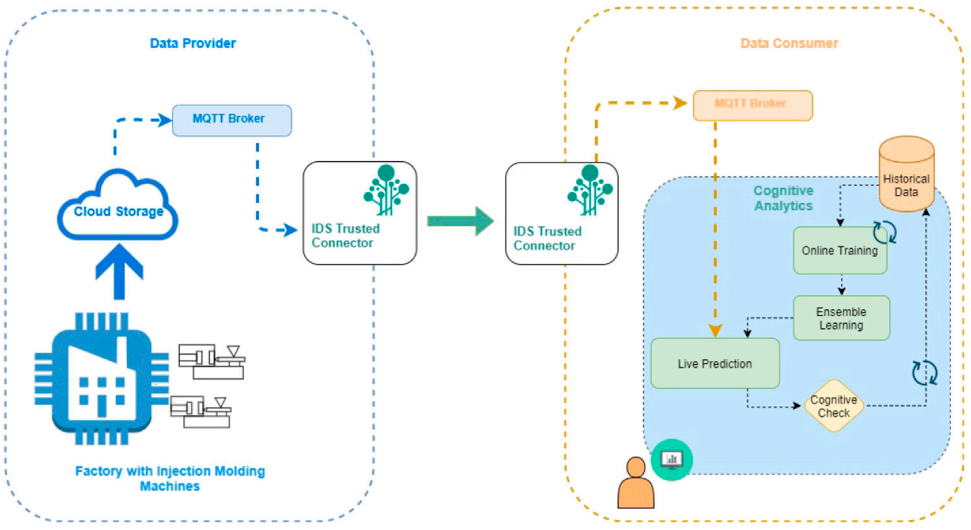


FIGURE 2 | Overview of the connection of the injection molding machines factory site with the proposed cognitive analytics.

connector is deployed on the factory site. The machines send data to the cloud infrastructure that is available to the factory and from there, the data are provided to IDS connector through an MQTT Broker. The factory cloud repository is the data provider of IDS architecture, whereas the cognitive analytics framework is the data consumer. A second IDS Trusted Connector was set up on consumer site alongside with a MQTT Broker in order to enable data exchange with the data provider. The IDS Trusted Connectors were selected because they offer an open platform which connects sensors with cloud infrastructures and other connectors in a secure and trusted way. In particular, the connectors are based on containers logic and provide apps isolation. They are isolated from each other and from the Internet. Furthermore, the connectors offer cross-enterprise authorization based on identity tokens. Another advantage of the data exchange between connectors is the ability to control and document the data usage. In addition to access control, the usage control allows for controlling data flows between apps and connectors. Based on the aforementioned advantages of IDS Trusted Connectors, they were selected as the ideal candidates to support the major requirement for secure transmission of the sensitive and private industrial data.

3.5 Cognitive Check

A cognitive mechanism is implemented at this point toward automated update of the prediction models. This mechanism triggers the retraining of the running models in two specific circumstances:

- the dataset's characteristics are changed
- the model's performance starts to downgrade

The new data that are constantly inserted in the prediction models are reassessed in order to capture possible variations compared to the historical data. The variance of the features of the historical datasets is stored and every new incoming data stream is compared with this value. If the new measurements are not statistically related to the training dataset, then the model training has to be repeated on the new dataset. In any case, the cognitive mechanism observes repeated measurements with variations until it finally triggers the retraining of the dataset, so as to eliminate accidental discrepancies of the machine live data.

The retraining is also activated by monitoring of the prediction model performance. An Initial Prediction Window (IPW) is determined at the training phase, which is a specific number of real-time predictions tested against the real ones when those are available. In case of the machines with labeled data, the real values are given and compared to the predicted ones. In case of the machines with unlabeled data, the result of major voting method substitutes the labels of data instances and is compared with the predicted results in order to extract the performance metrics. In both cases, a confusion matrix is created and the metrics, precision, recall, accuracy, and f-measure are calculated.

Based on the values of f-measure, the IPW is changed (increases or decreases) or remains the same. More specifically, a minimum and maximum value are defined for the IPW values, along with a threshold for the f-measure value. Starting from the maximum IPW value, f-measure is calculated for this window. If f-measure exceeds the defined threshold, the training model remains as it is, whereas IPW increases by 10 if f-measure is higher than 90%, decreases by 10 if f-measure is lower than 80%, and remains as it is if f-measure falls between 80 and 90%. This process is repeated until the IPW equals the minimum IPW. In case that f-measure falls behind the defined threshold, the retraining mode is triggered and the IPW value resets to the maximum value.

4 EXPERIMENTAL RESULTS AND COMPARISON WITH PRIOR WORK

The methodology described in the previous chapter refers to a dynamic and automatic system for real-time anomaly monitoring. The online training functionality is the basis for the live monitoring and is automatically triggered according to the cognitive mechanism. Since it is a live system, in order to evaluate its functionalities and performance, an indicative instance of training and testing of the prediction models is presented below. Furthermore, an experiment of the cognitive mechanism is presented at this chapter, showing the robustness of the proposed method. Finally, a comparison between our method and prior related work is performed.

4.1 Evaluation Metrics Overview

In order to assess our supervised models, we use the measures of precision, recall, accuracy, and f-measure, which are computed from the contents of the confusion matrix of the classification predictions. Because of the fact that we do not have binary classification, all the evaluation metrics are computed accordingly. From the confusion matrix true positive and false positive cases are denoted as TP and FP, while true negative and false negative are denoted as TN and FN, respectively. Precision is the ratio of predicted true positive cases to the sum of true positives and false positives and is given by the equation

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}.$$

Recall is the proportion of the true positive cases to the sum of true positives and false negatives and is given by the equation

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}.$$

Accuracy is the fraction of the total number of predictions that were correct and is given by the equation

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}.$$

Precision or recall alone cannot describe a classifier's efficiency. Therefore, f-measure is introduced as a combination of these two metrics. It is defined as twice the harmonic mean of precision and recall and is the metric we will be most referring to. The equation of f-measure is given below:

$$\text{f} - \text{measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}.$$

A value closer to one means better combined precision and recall of the classifier, whereas lower values imply worst accuracy or precision or both.

Accordingly, the unsupervised model assessment is performed by four clustering performance evaluation metrics: Silhouette Coefficient, Calinski–Harabasz index, Davies–Bouldin index, and Dunn index. Those are metrics for evaluating clustering algorithms following an internal evaluation scheme, where the metric result is based on the clustered data itself. The Silhouette Coefficient is an example of evaluation using the model itself (Rousseeuw, 1987). The Silhouette Coefficient for a single sample is given as

$$\text{Silhouette} = \frac{b - a}{\max(a, b)},$$

where a is the mean distance between a sample and all other points in the same class and b is the mean distance between a sample and all other points in the next nearest cluster. The Silhouette Coefficient for a set of samples is given as the mean of the Silhouette Coefficient for each sample. Higher Silhouette Coefficient scores indicate a model with better defined clusters.

Another evaluation metric, in case that the ground truth labels are not known, is the Davies–Bouldin index (Davies and Bouldin, 1979). The "similarity" between clusters is measured by this metric by comparing the distance between clusters with the size of the clusters themselves. The Davies–Bouldin index is specified as

$$\text{DB} = \frac{1}{K} \sum_{i=1}^{k} \max_{i \neq j} R_{ij},$$

where R_{ij} is the similarity measure defined as

$$R_{ij} = \frac{s_i + s_j}{d_{ij}}$$

for each cluster C_i for $i = 1, \ldots, k$ and its most similar one C_j:

- s_i is the average distance between each point of cluster i and the centroid of that cluster.
- d_{ij} is the distance between the cluster centroids i and j.

The lowest possible score is zero and values closer to zero suggest a better partition. Next is the Calinski–Harabasz index also known as the Variance Ratio Criterion (Caliński and Harabasz, 1974). The index is the ratio of the sum of the between-clusters dispersion and inter-cluster dispersion for all of them:

$$\text{CH} = \frac{\text{tr}(B_k)}{\text{tr}(W_k)} \times \frac{n_E - k}{k - 1},$$

where $\text{tr}(B_k)$ is trace of the between-group dispersion matrix and $\text{tr}(W_k)$ is the trace of the within-cluster dispersion matrix defined by

$$W_k = \sum_{q=1}^{k} \sum_{x \in C_q} \left(x - c_q\right)\left(x - x_q\right)^T$$

and

$$B_k = \sum_{q=1}^{k} h_q \left(c_q - c_E\right)\left(c_q - c_E\right)^T,$$

TABLE 3 | Averages of precision, recall, accuracy, and f-measure for 100 Monte-Carlo iterations for SVM-POLY classifier.

p	θ	Precision (%)	Recall (%)	Accuracy (%)	f-Measure (%)
2	30	75.24	88.92	78.53	81.51
2	36	77.82	85.91	79.24	81.67
2	42	77.54	86.27	79.32	81.67
2	48	78.35	86.98	79.54	82.44
2	56	78.78	86.11	79.23	82.28
3	30	75.39	87.92	79.21	81.17
3	36	76.41	87.69	79.11	81.66
3	42	76.12	87.77	78.92	81.53
3	48	78.01	87.34	78.24	82.41
3	56	78.09	86.92	77.88	82.27
4	30	77.98	85.43	77.87	81.54
4	36	77.15	86.13	77.46	81.39
4	42	77.26	86.24	77.79	81.50
4	48	78.65	86.71	77.92	82.48
4	56	78.92	87.01	78.01	82.77

TABLE 4 | Averages of precision, recall, accuracy, and f-measure for 100 Monte-Carlo iterations for SVM-RBF classifier.

C	σ	Precision (%)	Recall (%)	Accuracy (%)	f-Measure (%)
1,000	30	78.66	83.41	79.58	80.97
1,000	36	77.65	83.45	79.61	80.45
1,000	42	78.24	83.98	79.54	81.01
1,000	48	78.54	84.14	80.13	81.24
1,000	56	78.98	84.35	79.98	81.58
3,000	30	79.13	83.85	79.32	81.42
3,000	36	79.24	83.24	79.45	81.19
3,000	42	79.47	84.25	79.33	81.79
3,000	48	80.13	84.27	80.24	82.15
3,000	56	79.91	82.78	80.76	81.32
5,000	30	73.37	89.86	80.18	80.78
5,000	36	74.27	88.24	80.27	80.65
5,000	42	73.45	88.15	80.13	80.13
5,000	48	73.26	88.97	80.54	80.35
5,000	56	73.15	88.48	79.93	80.09

where C_q is the set of points in cluster q, c_q is the center of cluster q, c_E is the center of E, and n_q is the number of points in cluster q. The higher Calinski–Harabasz score implies a model with better defined clusters. Last is Dunn index (Dunn, 2008), another metric that aims to identify the compact sets of clusters and the well-separated ones. The metric is given by the following equation:

$$\mathrm{DI} = \frac{\min\limits_{1\le i\le j\le m} \delta\left(C_i, C_j\right)}{\max\limits_{1\le k\le m} \Delta_k},$$

where $\delta(C_i, C_j)$ is the distance between clusters C_i and C_j, and Δ_k is the intracluster distance within cluster C_k. The higher the Dunn index value is, the better the model performance is.

4.2 Training Simulation Results

The section of experimental results regarding online training is divided into three subsections. In the first subsection, the performance of all tested classifiers is presented, while in the second subsection the boosted version of the classifier with the best predictive performance among tested ones is presented. The third subsection presents results from the unsupervised training. Out of all experiments conducted in this research in order to test and evaluate the proposed methodology, some indicative results are given below in order to show the potential of the system and the attempt to create a compound solution for the injection molding machines. The evaluation of the system presented at this point focuses on both the functionalities of the proposed solution and the performance of the available algorithms presented in **Section 3**.

4.2.1 Nonboosted Version of Classifiers

In order to evaluate the predictive performance of tested classifiers, a series of 100 Monte-Carlo simulations was performed, for each parameter schema. The idea behind Monte-Carlo simulations is the generation of a large number of synthetic datasets that are similar to experimental data. In the case of time series the simulation setup of the match for Monte-Carlo realizations is 100-fold cross-validation. For SVM-POLY, parameter θ takes the values θ = (start = 30, end = 60, step = 6) and the polynomial degree p takes the values $p = (2, 5, 1)$. For SVM-RBF, parameter σ varies the same as θ and the constant C as C = (1,000, 7,000, 2,000). In **Tables 3, 4**, we present the simulation results of SVM-POLY and SVM-RBF classifiers, percentage averages for 100 Monte-Carlo iterations for precision, recall, accuracy, and f-measure. The classic BPN has a single hidden layer and the number of neurons varies as $n = (100, 200, 20)$. The simple decision tree was tested as is while the random forest has an ensemble of estimators = (20, 100, 20) decision trees. In **Tables 5, 6**, we present the simulation results of BPN and random forest classifiers, percentage averages for 100 Monte-Carlo iterations for precision, recall, accuracy, and f-measure. From all the simulation results presented in **Tables 3**-**6**, it is more than clear that random forest classifier outperforms SVM-POLY, SVM-RBF, and BPN

TABLE 5 | Averages of precision, recall, accuracy, and f-measure for 100 Monte-Carlo iterations for BPN classifier.

Neurons	Precision (%)	Recall (%)	Accuracy (%)	f-Measure (%)
100	72.28	90.21	80.52	80.26
120	72.37	89.25	80.24	79.93
140	72.56	89.76	80.52	80.25
160	72.89	89.91	80.59	80.51
180	73.24	90.73	81.11	81.05

TABLE 6 | Averages of precision, recall, accuracy, and f-measure for 100 Monte-Carlo iterations for random forest classifier.

Decision trees	Precision (%)	Recall (%)	Accuracy (%)	f-Measure (%)
20	92.24	93.47	91.24	92.85
40	92.37	94.13	91.78	93.24
60	93.91	93.71	91.25	93.81
80	93.63	94.61	91.51	94.12
100	93.24	94.28	91.39	93.76

TABLE 7 | Averages of precision, recall, accuracy, and f-measure for 100 Monte-Carlo iterations of AdaBoost on random forest classifier.

Parameters of weak learners	Precision (%)	Recall (%)	Accuracy (%)	f-Measure (%)
20, 40, 60, 80, 100	97.63	97.28	95.32	97.45

TABLE 8 | Unsupervised evaluation metrics on injection molding machine 5/6.

Algorithms	Injection machine	Silhouette	Davies–Bouldin	Calinski–Harabasz	Dunn
LOF	Machine 5	0.05	4.57	119.89	0.11
	Machine 6	0.11	2.83	400.44	0.03
K-means	Machine 5	0.15	3.29	774.63	0.05
	Machine 6	0.21	2.16	1,041.22	0.02
DBSCAN	Machine 5	0.05	4.80	1,456.17	0.02
	Machine 6	0.42	1.51	1,449.55	0.41
One-Class SVM	Machine 5	0.05	17.18	134.61	0.01
	Machine 6	0.01	18.52	98.59	0.01

for about 11–12%. Thus, random forest is the one classifier that will be promoted to test also in its boosted form.

4.2.2 Random Forest Boosted Version

In order to have a more clear view about the potential of random forest, we simulate different schemas of estimators and calculate again precision, recall, accuracy, and f-measure. In this simulation scenario, denoted hereafter as RF-Boost, five weak learners (or five random forest classifiers) were used where the estimator of each one of the weak learners is estimators = [40, 60, 80, 100, 120] decision trees. The simulation results of RF-Boost scenario are given in **Table 7**. Comparing **Tables 6, 7**, one can see that, with the RF-Boost scenario, the predictive performance is increased by 3–4% on f-measure, a fact that indicates the dominance of boosted form compared to any other predictive approach, tested here.

4.2.3 Unsupervised Learning Results

As mentioned before, two of the available injection molding machines lack ground truth values. The simulation results of the unsupervised training applied to these machines are shown in **Table 8**. The values of the table refer to both machines trained with historical data. According to the evaluation metric description in **Section 4.1** it seems that One-Class SVM is the weaker learner and LOF and DBSCAN have given better results in machines five and six, respectively. The challenge of the unlabeled data is to be well evaluated in order to use the corresponding models as fault detectors. It may be a weaker method compared to supervised evaluation but the system aims to give a handful solution in case where ground truth is missing and give accurate results in cooperation with the ensemble enhancement given by majority voting and also with the retraining module of the proposed methodology.

4.3 Cognitive Mechanism Testing

The prediction models generated by the online training (**Section 3.2**) are used in live prediction phase (**Section 3.4**) where real-time data are monitored and anomalies are detected. The cognitive mechanism operates at the same time as live prediction and triggers the retraining of a prediction model when needed. In order to test this feature, the performance of models was recorded as live prediction and cognitive check are operating. Specifically, four prediction models are monitored: decision tree and random forest models for the supervised learning of labeled data and DBSCAN and K-means models for the unsupervised learning of unlabeled data. The accuracy evaluation metric is recorded for the supervised learning and the silhouette score for the unsupervised. The model performance was recorded for as long as it takes for the cognitive mechanism to trigger 200 retraining times of the models.

Figure 3 shows some indicative results of this testing. It illustrates the diagrams of the evaluation metric throughout 200 retraining times of a single model (accuracy for supervised and silhouette score for unsupervised learning). The diagrams indicate that the models' performance is maintained in high levels after the model retraining: decision tree classifier's accuracy does not fall under 0.997 and random forest under 0.992 and DBSCAN and K-means' silhouette scores are kept over 0.2 and 0.14, respectively. The model retraining is triggered by variations noticed in the real-time data compared with the historical ones, so it is crucial that it will be accomplished the time that is being triggered, regardless of the results that it will induce.

As a follow-up to the above diagrams, the times where the model was improved after retraining were calculated. **Table 9** shows that most of the times the execution of retraining improves the performance of the model. The aim of the retraining module is to automatically update the predictive models when their performance diverges. These indicative results explain the need of the complete system to be updated occasionally in order to be able to maintain high quality and accuracy in the anomaly detection pipeline.

4.4 Comparison with Prior Work

In order to support our proposed methodology, we present a determinate comparison between our methodology and other works from literature. The comparison concerns the studies that deal with injection molding machines as this is the core of our research and focuses on aspects of each proposed methodology, since the results of each work are disparate or

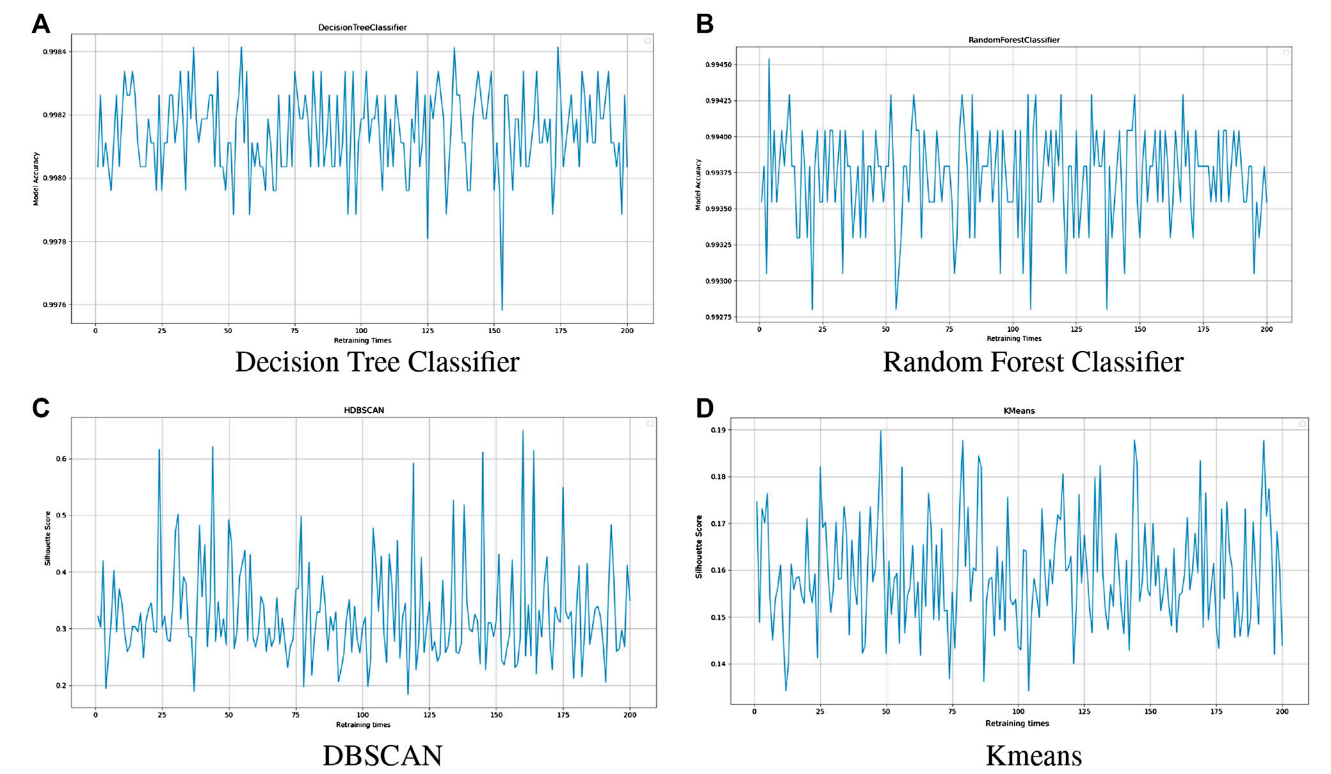


FIGURE 3 | Cognitive mechanism performance monitoring. The horizontal axis of the graphs shows the times of retraining and the vertical axis shows the accuracy **(A,B)** and silhouette score **(C,D)** of the model.

TABLE 9 | The percentage of retraining times that accuracy improvement is noticed.

Algorithm	Machine	Retraining improvement (%)
Decision tree classifier	Injection molding machine 1	63
Random forest	Injection molding machine 2	63
K-means	Injection molding machine 5	75
DBSCAN	Injection molding machine 6	73

unavailable. The first study includes an injection molding machine's use case and explores the available data while the machine operates (Gatica et al., 2016). The study extracts normal behavior models and notices deviations from the expected behavior. The authors use trend analysis functions to predict already known failures and achieve reduction of machine downtime.

The second study deals with anomaly detection on streaming data applied to injection molding machines (Jankov et al., 2017). A sliding window observes the streaming data and finds clusters by using K-means algorithm. The clusters are used for training of a Markov model for the window. New models are trained as the window slides over new data. Anomalies are detected in the streaming data by calculating transition probability and comparing it with a threshold. The advantage of Jankov et al. (2017) work is the computational capabilities of the system which is augmented by real-time parallel task distribution.

Park et al. (2016) address the problem of machine condition monitoring by identifying the injection molding operational parameters. Statistical analysis is applied to these parameters in order to distinct the most significant ones. Real-time data series are monitored by prediction models and the results are evaluated by Nelson rules. The method detects abnormal patterns of the parameters and identifies the machine parts where maintenance actions should aim.

Our proposed method follows a similar approach to the aforementioned works. The aim is to investigate abnormal operations in the injection molding process starting with data analysis and resulting in prediction models that determine anomalies. In contrast with the studies above, we proposed a methodology which can handle both labeled and unlabeled data and also address the challenge of unknown errors in case of machine abnormal behavior. Additionally, the other works derive prediction models using one specific analytical method, but the current study includes multiple classification and clustering learners for the generation of prediction models. From the set of trained models, the one with higher performance will be used for real-time anomaly monitoring. Also, there is the capability of meta-learning as described in **Section 3.3**.

The distinguishing feature of the current work though is the constant updating of the system's prediction models through cognitive retraining. Except Jankov et al. (2017) work which trains new prediction models as they cross by data streams, the remaining studies do not focus on the potential changes that can be noticed in live data or the possible degradation of the prediction methods. This is the core of the cognition aspect of the current work which achieves steady performance of the real-time anomaly detection system.

5 CONCLUSIONS

In this paper a cognitive analytics application is presented, focusing on predictive maintenance applied to injection molding machines. A complete solution was described in detail including different stages of training of historical data, live prediction on real-time data, and automated retraining which aims to keep the prediction process up to date. The proposed solution manages both labeled and unlabeled datasets and applies ensembles methods to top of individual supervised and unsupervised learners. The generated prediction models receive real-time data streams and perform anomaly detection on the features of the injection molding machine measurements. A cognitive mechanism was developed and tested, which monitors the dataset changes, on the one hand, and the model performance, on the other hand, and constantly updates the predictive models.

The main findings of our research are summarized below:

- The proposed solution achieves combining different training methods and detecting faults in different machines, located in the same factory site.
- Ensemble methods can enhance the prediction models' performance results.
- Automatic updating of trained models addresses the problem of possible deviations of new incoming machine data or potential prediction models' degradation.
- High model performance is preserved in real-time anomaly detection and data monitoring through automatic triggering of model retraining.

As a result of these assets, the presented method can constitute an assisting tool for the decision support system of factory sites facilitating injection molding machines, in order to prevail failures in production and downtime of machines.

Current ongoing work is implementing the creation of user interfaces for the proposed real-time anomaly detection methodology. Advanced visualizations are incorporated, offering an enhanced user experience and a thorough view of raw data, processed and clean data, model training, evaluation, and results, as well as real-time monitoring. Two user views are set up: the data scientist view and the regular end user. The data scientist can choose parameters and methods for online training which will operate the live monitoring for the regular end user. The anomalies are detected and visualized so as the predictive maintenance manager can make the necessary decisions in case of machine abnormalities.

Future work will concentrate on applying the presented methodology to different machine data of the Industry 4.0 domain and investigate a generic cognitive analytics framework for predictive maintenance. The development of more learning techniques is being considered as a next step, especially regarding the field of ensembled methods. Lastly, there is definitely a room for improvement in the unsupervised learning area regarding evaluation and meta-learning processes.

DATA AVAILABILITY STATEMENT

The data analyzed in this study are subject to the following licenses/restrictions: The dataset is private and anonymized. Requests to access these datasets should be directed to vrousop@iti.gr.

AUTHOR CONTRIBUTIONS

This paper is a joint work from all listed authors. In particular, VR and AN contributed to conceptualization and writing of the first draft of the document. VR is also the main contributor in presented framework's development. Furthermore, TV wrote the section related to ensemble methods and contributed to framework's design alongside VR and AN. DI and DT supervised the work and provided the necessary design guidelines. All the authors contributed to changes and corrections before the final submission.

ACKNOWLEDGMENTS

This project has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No. 780732–BOOST 4.0. This paper reflects only the authors' views and the Commission is not responsible for any use that may be made of the information it contains. Furthermore, we thank Bas Tijsma from Philips Drachten for providing us with anonymized data that we used in the current study.

REFERENCES

Bannat, A., Bautze, T., Beetz, M., Blume, J., Diepold, K., Ertelt, C., et al. (2011). Artificial cognition in production systems. *IEEE Trans. Autom. Sci. Eng.* 8, 148–174. doi:10.1109/TASE.2010.2053534

Boza, A. S., Guerra, R. H., and Gajate, A. (2011). Artificial cognitive control system based on the shared circuits model of sociocognitive capacities. A first approach. *Eng. Appl. Artif. Intell.* 24, 209–219. doi:10.1016/j.engappai.2010.10.005

Breiman, L. (1999). Random forests – random features. *Nature* 567, 5–32. doi:10.1023/A:1010933404324

Calinski, T., and Harabasz, J. (1974). A dendrite method for cluster analysis. *Commun. Stat. Theory Methods.* 3, 1–27. doi:10.1080/03610927408827101

Davies, D. L., and Bouldin, D. W. (1979). A cluster separation measure. *IEEE Trans. Pattern Anal. Mach. Intell.* 1, 224–227. doi:10.1109/tpami.1979.4766909

Domingues, R., Buonora, F., Senesi, R., and Thonnard, O. (2016). "An application of unsupervised fraud detection to passenger name records," in 2016 46th annual IEEE/IFIP international conference on Dependable Systems and Networks Workshop (DSN-W), Toulouse, France, June 28–July 1, 2016.

Domingues, R., Filippone, M., Michiardi, P., and Zouaoui, J. (2018). A comparative evaluation of outlier detection algorithms: experiments and analyses. *Pattern Recogn.* 74, 406–421. doi:10.1016/j.patcog.2017.09.037

Dunn, J. (2008). Well-separated clusters and optimal fuzzy partitions. *Cybern. Syst.* 4, 95–104. doi:10.1080/01969727408546059

Freund, Y., and Schapire, R. (1995). *A desicion-theoretic generalization of on-line learning and an application to boosting*. New York, NY: Springer, 23–37.

Galetsi, P., Katsaliaki, K., and Kumar, S. (2020). Big data analytics in health sector: theoretical framework, techniques and prospects. *Int. J. Inf. Manag.* 50, 206–216. doi:10.1016/j.ijinfomgt.2019.05.003

Gatica, C. P., Koester, M., Gaukstern, T., Berlin, E., and Meyer, M. (2016). "An industrial analytics approach to predictive maintenance for machinery applications," in 2016 IEEE 21st international conference on Emerging Technologies and Factory Automation (ETFA), Berlin, Germany, September 6–9, 2016, 1–4.

Iarovyi, S., Lastra, J. L. M., Haber, R., and del Toro, R. (2015). "From artificial cognitive systems and open architectures to cognitive manufacturing systems," in 2015 IEEE 13th international conference on Industrial Informatics (INDIN). Cambridge, MA, July 22–24, 2015, 1225–1232.

Jankov, D., Sikdar, S., Mukherjee, R., Teymourian, K., and Jermaine, C. (2017). "Real-time high performance anomaly detection over data streams: grand challenge," in Proceedings of the 11th ACM international conference on distributed and event-based systems, New York, NY, June 19–23, 2017 (Association for Computing Machinery), DEBS '17), 292–297.

Jung, H., and Lease, M. (2012). Evaluating classifiers without expert labels. arXiv [preprint]. Available at: aeXiv:1212.0960v1 (Accessed December 5, 2012).

Lee, S. J., and Siau, K. (2001). A review of data mining techniques. *Ind. Manag. Data Syst.* 101, 41–46. doi:10.1108/02635570110365989

Nath, S. V., and Behara, R. S. (2003). "Customer churn analysis in the wireless industry: a data mining approach," in Annual meeting of the Decision Sciences Institute, Washington, DC, November 22–25, 2003, 505–510.

Otto, B., and Jarke, M. (2019). Designing a multi-sided data platform: findings from the international data spaces case. *Electron. Mark.* 29, 561–580. doi:10.1007/s12525-019-00362-x

Otto, B., Lohmann, S., Steinbuss, S., and Teuscher, A. (2019). *IDS reference architecture model version 3.0*. Dortmund: International Data Spaces Association.

Park, C., Moon, D., Do, N., and Bae, S. M. (2016). A predictive maintenance approach based on real-time internal parameter monitoring. *Int. J. Adv. Manuf. Technol.* 85, 623–632. doi:10.1007/s00170-015-7981-6

PLANTCockpit (2012). Plantcockpit white-paper. Available at: https://cordis.europa.eu/docs/projects/cnect/8/260018/080/deliverables/001-PLANTCockpitD33V10.pdf (Accessed September 3, 2012).

Rojko, A. (2017). Industry 4.0 concept: background and overview. *Int. J. Interact. Mob. Technol.* 11, 77. doi:10.3991/ijim.v11i5.7072

Rousopoulou, V., Nizamis, A., Giugliano, L., Haigh, P., Martins, L., Ioannidis, D., et al. (2019). "Data analytics toward predictive maintenance for industrial ovens: a case study based on data analysis of various sensors data," in Advanced information systems engineering workshops, 83–94.

Rousseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.* 20, 53–65. Workshops, Rome, Italy, June 3–7, 2019. doi:10.1016/0377-0427(87)90125-7

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature* 323, 533–536. doi:10.1038/323533a0

Salamanis, A., Kehagias, D. D., Filelis-Papadopoulos, C. K., Tzovaras, D., and Gravvanis, G. A. (2016). Managing spatial graph dependencies in large volumes of traffic data for travel-time prediction. *IEEE Trans. Intell. Transport. Syst.* 17, 1678–1687. doi:10.1109/tits.2015.2488593

Schapire, R. E., and Freund, Y. (2012). *Boosting: foundations and algorithms (adaptive computation and machine learning series)*. Cambridge, MA: MIT Press.

Vatrapu, R., Mukkamala, R. R., Hussain, A., and Flesch, B. (2016). Social set analysis: a set theoretical approach to big data analytics. *IEEE Access.* 4, 2542–2571. doi:10.1109/access.2016.2559584

Zaeh, M. F., Beetz, M., Shea, K., Reinhart, G., Bender, K., Lau, C., et al. (2009). *The cognitive factory*. London: Springer, 355–371.

Zhang, Z., Han, H., Cui, X., and Fan, Y. (2020). Novel application of multi-model ensemble learning for fault diagnosis in refrigeration systems. *Appl. Therm. Eng.* 164, 114516. doi:10.1016/j.applthermaleng.2019.114516

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

frontiers in Computer Science

ORIGINAL RESEARCH
published: 27 November 2020
doi: 10.3389/fcomp.2020.578469

Implementation and Transfer of Predictive Analytics for Smart Maintenance: A Case Study

Sebastian von Enzberg[1*], *Athanasios Naskos*[2], *Ifigeneia Metaxa*[2], *Daniel Köchling*[3] *and Arno Kühn*[1]

[1] *Research Department Product Engineering, Fraunhofer Institute for Mechatronic Systems Design, Paderborn, Germany,* [2] *ATLANTIS Engineering SA, Thessaloniki, Greece,* [3] *BENTELER Automobiltechnik GmbH, Paderborn, Germany*

Smart maintenance offers a promising potential to increase efficiency of the maintenance process, leading to a reduction of machine downtime and thus an overall productivity increase in industrial manufacturing. By applying fault detection and prediction algorithms to machine and sensor data, maintenance measures (i.e., planning of human resources, materials and spare parts) can be better planned and thus machine stoppage can be prevented. While many examples of Predictive Maintenance (PdM) have been proven successful and commercial solutions are offered by machine and part manufacturers, wide-spread implementation of Smart Maintenance solutions and processes in industrial production is still not observed. In this work, we present a case study motivated by a typical maintenance activity in an industrial plant. The paper focuses on the crucial aspects of each phase of the PdM implementation and deployment process, toward the holistic integration of the solution within a company. A concept is derived for the model transfer to a different factory. This is illustrated by practical examples from a lighthouse factory within the BOOST 4.0 project. The quantitative impact of the deployed solutions is described. Based on empirical results, best practices are derived in the domain and data understanding, the implementation, integration and model transfer phases.

Keywords: smart maintenance, predictive analytics, model transfer, industrial data science, best practices

OPEN ACCESS

Edited by:
Oscar Lazaro,
Innovalia Association, Spain

Reviewed by:
Luca Davoli,
University of Parma, Italy
Laura Belli,
University of Parma, Italy

***Correspondence:**
Sebastian von Enzberg
sebastian.von.enzberg@iem.fraunhofer.de

Specialty section:
This article was submitted to Mobile and Ubiquitous Computing, a section of the journal Frontiers in Computer Science

Received: *30 June 2020*
Accepted: *14 October 2020*
Published: *27 November 2020*

Citation:
von Enzberg S, Naskos A, Metaxa I, Köchling D and Kühn A (2020) Implementation and Transfer of Predictive Analytics for Smart Maintenance: A Case Study. Front. Comput. Sci. 2:578469. doi: 10.3389/fcomp.2020.578469

1. INTRODUCTION

1.1. Process Models for Implementation of Predictive Maintenance

A range of process models describe the software development and implementation process. A widely used process model for the domain of data analytics is the cross-industry standard process for data mining (CRISP-DM, Shearer, 2000). It describes an iterative process starting with domain and data understanding, data preparation, modeling, evaluation, and deployment. It has already been subjected to adaptions and advancement, especially in the context of Industrial Data Science, e.g., by (Reinhart, 2016).

When implementing Predictive Maintenance (PdM) solutions, an important consideration is the overall use case context in which data analytics is integrated. For example, the fault prediction of an electric engine is embedded into a manufacturing process where the electric engine serves a certain function (e.g., driving a belt for transport of material), the malfunction is accompanied by effects (e.g., stop of material transport), as well as consecutive maintenance processes (e.g., repair of the engine). While this may be covered by the domain understanding, an explicit use case analysis and definition phase is necessary for a benefit- and business-driven selection of a use case and its further development. In addition to the original deployment of a PdM solution, a transfer to similar scenarios and other plants is desired (e.g., fault prediction for the engines of other transport belts).

Figure 1 summarizes our proposed overall business-driven process for the implementation of predictive analytics. It can be divided into three main phases: Use Case Analysis, Proof of Concept, and Deployment. Next to the process-oriented view, the implementation can be divided into four interdisciplinary, functional layers of data analytics according to Kühn et al. (2018). It offers different views of the implementation process, covering the Use Case, Data Analytics, Data Pools, and Data Sources view. The process steps are integrated into the layer model in **Figure 2**.

In this article, we analyze the PdM implementation process with a strong real-world application and business-oriented focus. This process originates from the implementation experiences of the lighthouse factory of BENTELER Automotive within the BOOST 4.0—Big Data for Factories project. The BOOST 4.0 project is an EU-funded Innovation Action aiming to improve the competitiveness of the European manufacturing industry, to introduce Industry 4.0 technologies and to provide the necessary tools for obtaining the maximum benefit of Big Data.

We have gathered and grouped the best practices derived from the followed process. It should be mentioned that our goal is to take advantage of these observations, in our effort to bridge the gap between research work and industrial application.

The article is divided into four main sections, covering the domain understanding and use case definition (section 2), data infrastructure (section 3), data understanding and modeling (section 4), as well as the process integration and transfer of models (section 5). We summarize the best practices and conclude our observations in sections 6 and 7, respectively.

1.2. State of the Art

In recent years, technologies for big data management and processing as well as algorithms for fault and anomaly detection have matured to allow industry-grade application of smart maintenance. While smart maintenance features are offered by individual component providers of production systems,

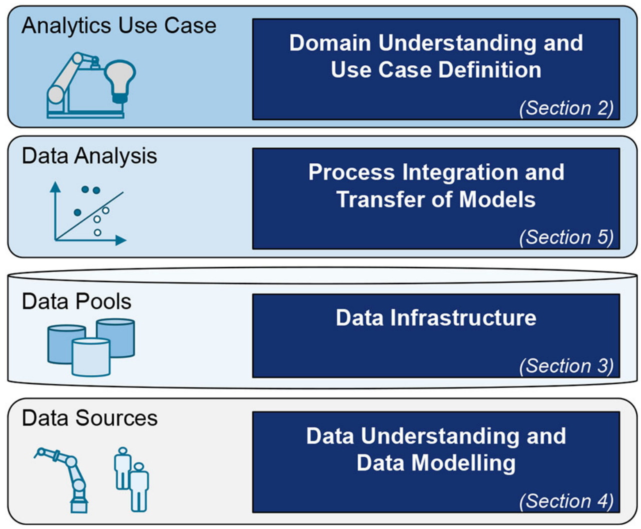


FIGURE 2 | Integration of process steps into the layer model for development of data analytics applications and overview of the best practice contributions in this paper.

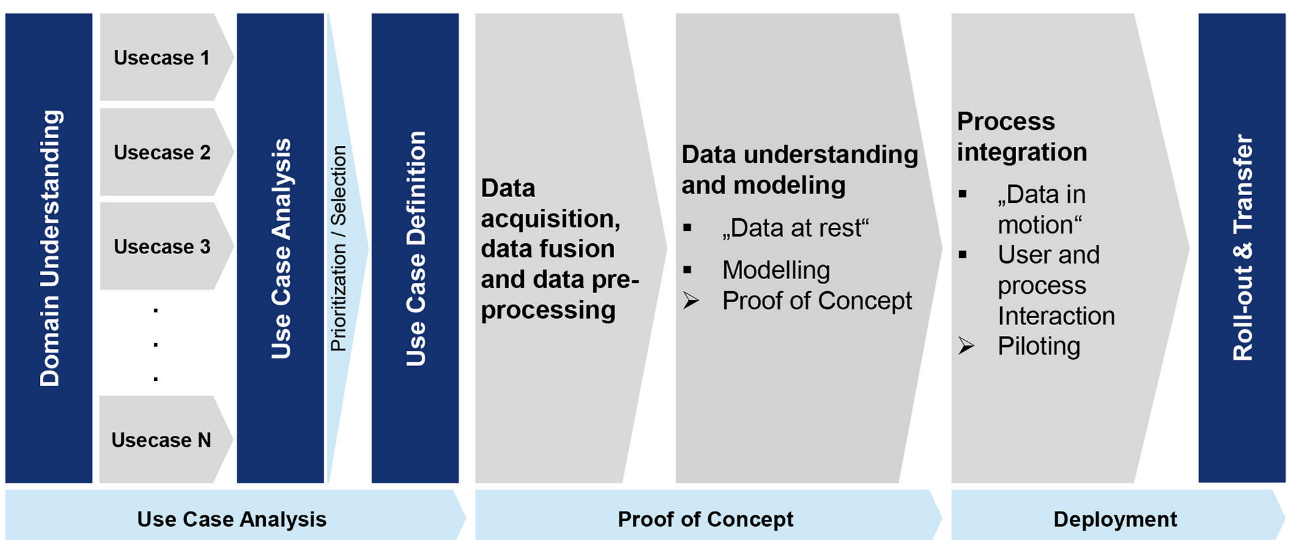


FIGURE 1 | Overall process for the implementation of predictive analytics.

widespread adoption of smart maintenance in manufacturing is still limited. Many companies cannot identify good business cases. High initial investments, as well as insufficient availability of mature or ready-to-use solutions, keep companies from the holistic integration of smart maintenance within the company. A stream-lined implementation and deployment process, along with easily transferable prediction models are key to the wide-spread application of smart maintenance within a company. It allows the transfer of a solution to different factories or different parts of a company, or even to other companies with similar settings.

The two-part study by Bokrantz et al. (2020b) and Bokrantz et al. (2020a) acknowledges the lack of empirically driven, conceptual work for smart maintenance. By means of an empirical study, it conceptualizes four main smart maintenance dimensions: data-driven decision making, human capital resources, internal integration and external integration. The conceptual framework offered by Zheng et al. (2018) ranges from sensors, data collection, and analytics to decision making. These framework dimensions are viewed in context of real-world scenarios, naming it as an important research aspect in Industry 4.0. It does not offer a specific guidance for smart maintenance.

Moens et al. (2020) offer an Industrial Internet of Things (IIoT) framework, in which smart maintenance solutions are embedded. They expose robustness and scalability of solutions as well as the availability of well-trained machine learning models for fault recognition as major challenges to be addressed. Bumblauskas et al. (2017) describe a decision support system as a smart maintenance framework. It is based on corporate big data analytics with integrated anomaly and fault detection methods. In Uhlmann et al. (2019), the process integration aspect of smart maintenance is focused. The authors propose an assistance systems, which helps to embed smart maintenance solutions integrally into service processes.

In summary, a range of recent work has focused on concepts for implementation of smart maintenance, rather than entirely technically-driven work. Dimensions that need to be addressed range from sensor data, data infrastructure to analytics and process integration of smart maintenance.

2. DOMAIN UNDERSTANDING AND USE CASE DEFINITION

The practical examples presented in this work are obtained from real industrial use cases of the BENTELER automotive lighthouse factory of the BOOST4.0 project. BENTELER produces and distributes safety-relevant products, serving customers in automotive technology, the energy sector and mechanical engineering. The production of such plants employs complex machinery to a large extent with several mechanical and hydraulic systems, which entail frequent and/or periodic maintenance. A thorough understanding of the problem domain is precondition to defining valuable use cases, data understanding and successful modeling and process integration. These steps are detailed in the following paragraphs.

2.1. Domain Understanding

The understanding of the application domain is of equal importance as the development of the actual smart maintenance solution. The domain and business understanding is a multi-phase iterative process, comprised by interviews with the maintenance engineers, interviews with the industrial IT and automation experts and knowledge transfer.

Initially, organizing interviews with the domain experts, i.e., maintenance engineers, is a necessary prerequisite in order to understand the underlying systems and behavior of signals, which are later considered in data cleaning, data pre-processing and algorithm development. Semi-formalized methods such as CONSENS (Conceptual Design Specification Technique for the Engineering of Complex Systems, Gausemeier et al., 2009) have been used in our use cases, giving effective structure diagrams for the machinery under consideration. For the use case of the scrap belt, an example of the effective structure of the scrap belt is given in **Figure 3**.

Understanding of Data and Infrastructure has been accomplished via interviews with industrial IT and industrial automation experts, as a necessary prerequisite in order to understand details of data collection, which directly affect data properties. This includes knowledge about data collection at OPC-UA level from industrial automation experts, which affects e.g., naming of measurements, timing of acquisition, etc. Also, knowledge from industrial IT experts allows a better understanding of data transfer to databases and resulting effects, e.g., generation and synchronization time stamps, data quantization. This information is important for correct selection and parameterization of algorithms for data pre-processing and filtering. Insights from data understanding and data preparation have been fed back to the domain and business understanding in weekly meetings with maintenance and IT experts.

In various workshops and online conferences, the domain, business and data understanding has been transferred between the end users and the technology providers. For the technology providers, a profound technical understanding of the production line at BENTELER is essential to allow goal-driven algorithm development. This is accelerated by proper documentation of domain understanding. The effective structure as shown in **Figure 3**, allows for successful transfer of domain and business know-how. For example, an observation of a maintenance incident is unambiguously assigned to a system component (red dot in **Figure 3**). Root Causes can be traced within the system and relevant data sources can be identified and located (white dot **Figure 3**). Preliminary algorithm results can be interpreted directly during their development, leading to quicker iterations. However, regular feedback from domain experts is still necessary. Weekly meetings of the core partners have allowed for continuous identification of bottlenecks in the knowledge transfer process.

2.2. Use Case Selection and Definition

In order for a Smart Maintenance solution to be evaluated and, if suitable, adopted in production line to the optimal degree, the costs and benefits need to be considered. The baseline cost is composed by the installation and operation of the IT

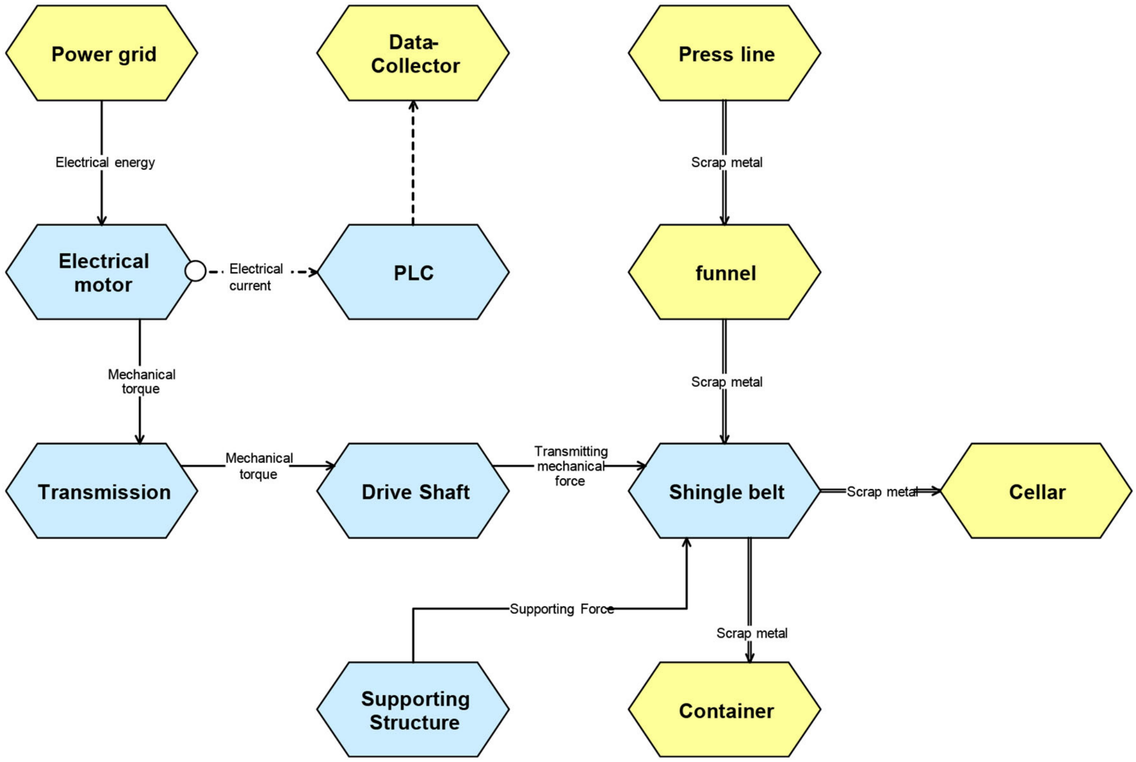


FIGURE 3 | Structure of the scrap belt.

infrastructure (hardware and software for Big Data storage and processing). This represents a significant investment, which also facilitates a wide range of possible subsequent benefits, it is thus a mostly strategic decision. Operative decision-making focuses cost-benefit considerations on a use case basis. The necessary steps are use case identification, use case selection and use case definition.

For *use case identification*, we propose a workshop-based approach with domain experts, that can be held in conjunction with the collection of domain understanding. A high-level overview of the systems is used as a guide for the identification process. This can be a schematic, floor-plan or domain-specific descriptions as wiring diagrams or hydraulic plans. The effective structure as shown in **Figure 3** is useful, as it can be easily understood by various stake holders. In **Figure 4**, we show how two possible use cases for data analytics, the prediction of spikes in motor current and the correlation of machine outage and product number, are identified alongside the effective structure. Additionally, details about the use cases can be directly recorded on a plot (e.g., typical failures and failure propagation, are marked in red on **Figure 4**). Other systematic approaches can also be exploited for use case identification, e.g., Failure Mode and Effects Analysis (FMEA).

The *use case selection* is based on a qualitative assessment of use cases. For each identified use case, two dimensions are considered: strategic value and possible benefit and simplicity of implementation and realization. As shown in **Figure 5**, this allows a simple overview and selection of most relevant use cases. The two use cases "oil leakage at hydraulic press" and "outage of scrap belt" have been identified for the BENTELER automotive lighthouse factory, as they rank high in both observed dimensions. Furthermore, a clustering allows the identification of neighboring use cases that be considered subsequently. In the example shown, a Machine-Health-Index is an overarching application scenario for several use cases.

Lastly, in the *use case definition* a more detailed but concise description is created, which allows the definition of the subject for further planning or a quantitative cost-benefit calculation. This work focuses on two practical use cases dealing with the hydraulic system of press and a conveyor belt moving scrap produced by presses. This step by step process has been adopted, in order to provide of a Proof of Value (POV) approach, which is necessary before considering an overarching Smart Maintenance solution, spanning more machines and failures of production lines.

2.2.1. The Scrap Belt Use Case

The scrap belt is connected to several lines and runs underground the BENTELER factory hall. Any scrap metal accumulated during the production process is placed onto the scrap belt. Scrap metal parts are then transported from the production line to a scrap metal container, and then to recycling. The proper functionality of the scrap belt is crucial to production, since a halt of the scrap belt means a potential halt of several production lines.

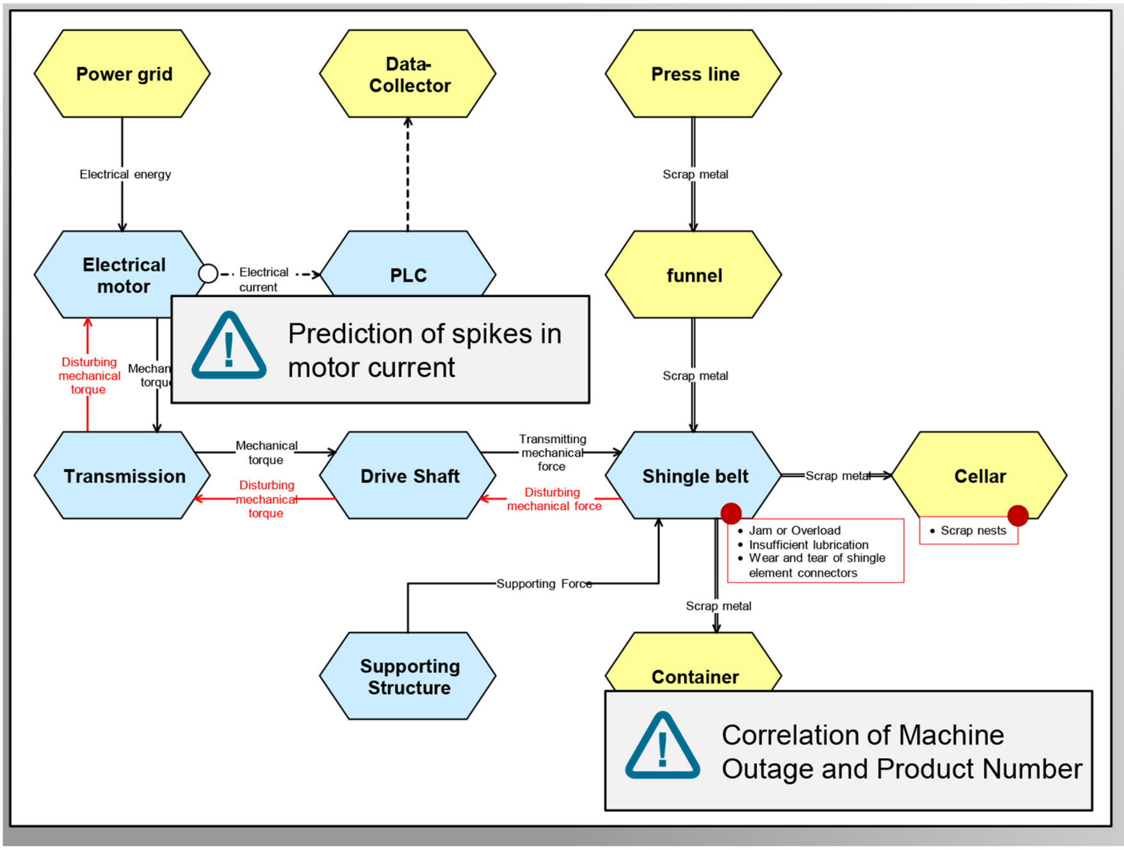


FIGURE 4 | Identification of Use Cases along the machine structure.

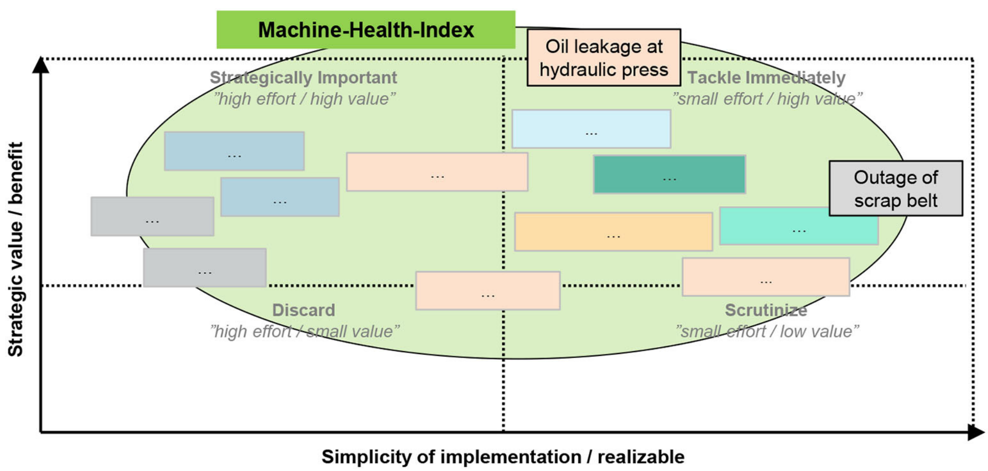


FIGURE 5 | Selection of Use Cases with a portfolio. Colored boxes represent Use Cases (partly anonymized), where similar colors represent logic groups of Use Cases. Additionally, clusters can be found (e.g., all Use Cases contribute to "Machine-Health-index," marked green). The two example Use Cases "oil leakage at hydraulic press" and "outage of scrap belt" are located in the upper right quadrant, thus being most simple to implement, yet yielding most benefit.

The focus of our study is to early detect pile-up of scrap metal and, thus, allow for time to take counter-measures in order to prevent a halt of the scrap metal belt. Even in the case of a complete halt, maintenance can be triggered much faster due to the continuous condition monitoring. In addition, diagnostic algorithms can provide maintenance or repair advice, suggesting

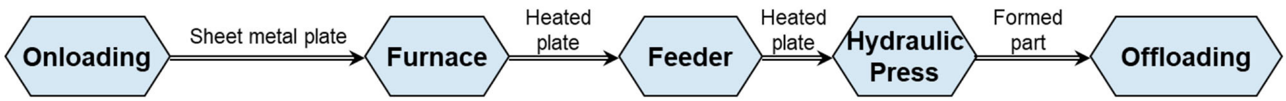


FIGURE 6 | Hot forming press process step.

possible causes of the failure. This significantly limits the manual fault diagnosis in the underground tunnel and allows a faster restoring of the regular belt functionality.

2.2.2. The Hydraulic Press Use Case

The hydraulic press use case is a complex hot forming line consisting of five consecutive process steps as presented in **Figure 6**. Its main task is the stamping of a sheet metal into a three-dimensional shape. The metal is heated before stamping and rapidly cooled down during stamping. This causes the material to be hardened, which is important for structural components for the automobile industry.

Our study focuses on the early detection of oil leakage occurrences. Despite the fact that, typically, oil is mostly stored in large tanks equipped with oil level sensors, oil leakage detection is a challenging problem due to the continuous movement of oil across the machinery equipment parts. Such movement results in frequent increases and decreases of oil level. Therefore, and somehow counter-intuitively, simply monitoring the oil level is not adequate to provide concrete evidence about oil leakage.

This business process is heavily affected by the installment of predictive and smart maintenance processes. The main objective of smart maintenance algorithms is the detection (condition monitoring) and prediction of oil leakages. Based on our results, maintenance processes can be triggered much faster or even in advance. Maintenance and repair activities can be planned more efficiently, and manual diagnosis is prevented.

2.3. Best Practices

It has been verified by the current work that domain understanding is important in order to build goal driven solutions. Interviews with domain experts (i.e., maintenance engineers, industrial IT, automation experts) are the most direct means for knowledge transfer. Semi-formalized methods like CONSENS assist the knowledge transfer process providing visualizations (effective structure diagrams) of the machinery.

The utilization of reactive (i.e., fault detection) and proactive (i.e., failure prediction) monitoring approaches can potentially increase the speed of reaction to crucial maintenance issues, enabling the prescriptive maintenance in which the maintenance and repair activities can be planned a priory.

3. DATA INFRASTRUCTURE

Efficient data handling is a crucial factor on the application of the smart maintenance approaches and on the model and knowledge transfer processes. The amount of sensor and production data produced on the BENTELER plants on daily basis is of the size of Big Data. In order to allow the analysis of this massive amount of data BENTELER is deploying a common policy of data handling for all distinct plants.

Each plant has a local time series database infrastructure, where through OPC-UA and other proprietary tools and protocols, all the produced data are persisted. For the development and testing of new smart maintenance solutions, BENTELER has created a remotely hosted cluster infrastructure, orchestrated by a containerized applications management software, called Developers Space. On this containerized infrastructure, the data of each distinct BENTELER plant can be mirrored on a local time series database, **Figure 7**, using an advanced distributed data streaming platform.

Deploying a smart maintenance solution in the Developers Space virtually deploys the same solution in all the connected with the Developers Space plants. If the deployed solution satisfies specific quality and performance aspects it can either remain deployed in the Developers Space, or if it is necessary it can be deployed on a local containerized infrastructure in one of the plants, in order to place the solutions closer to the use case that it monitors.

The components of our smart maintenance platform as deployed in the Developers Space are presented in **Figure 7**. The platform uses a micro-service architecture, where all the micro-services communicate with each other through a common event bus. A Dashboard allows the user to instantiate the services and to visualize the results. The main components of the platform are:

- DataProvider: Responsible to communicate with the time series database in order to fetch the latest sensorial data.
- Detection: A Fault Detection service to detect anomalies is the sensor measurements.
- Prediction: A Failure Prediction service to predict prominent failures based on the current sensor measurements.
- Fusion: Responsible to combine the output from multiple either Detection or Prediction services into a single result based on a pre-specified strategy.
- Reporter: Reports the results of the data analytics services, for further processing and visualization.

The utilized micro-service architecture enables the distributed deployment of the platform into decoupled entities, which are developed and evolve independently to each other providing a flexible smart maintenance solution.

3.1. Best Practices

A central data infrastructure, as the one presented in this section (i.e., Developers Space), enables the faster testing of prototypes in a wider range of use cases. After the successful evaluation phase in the sandboxed environment, the application may be moved closer to the application scenario using fog computing

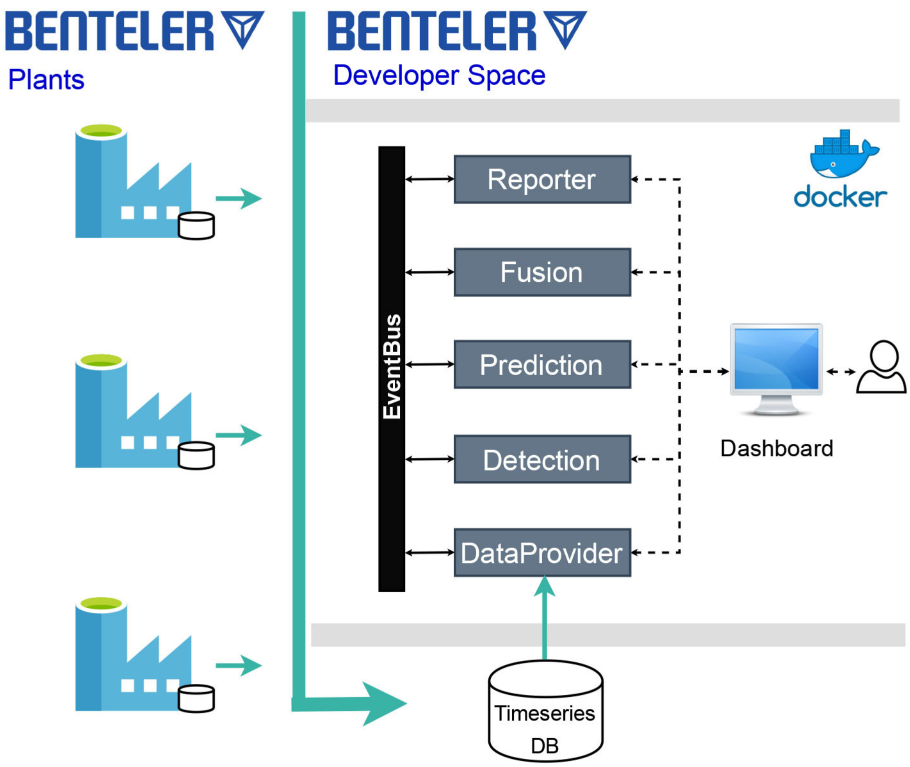


FIGURE 7 | BENTELER data infrastructure for smart maintenance.

and edge devices in order to achieve faster data transfer for critical cases. If the requirement for instant data transfer is relaxed, then the application may continue being deployed in the central infrastructure facilitating its maintenance.

A containerized environment in the central infrastructure assists the deployment of multiple solutions utilizing the minimum hardware resources. Another advantage is that if a solution tested to work in the central data infrastructure like in our case, it is also going to work on premises in other BENTELER plants in containerized environments.

The micro-service architecture of the software solution is the most appropriate for containerized environments, as it allows the distribution of the work among different containers providing the same functionality. Critical parts of the software solution that demand instant data transfer rates can be deployed on premises in fog nodes (i.e., node of fog computing), while the software components that need more computation resources and allow more relaxed data transfer rates can be deployed in a remote location like the Developers Space.

4. DATA UNDERSTANDING AND MODELING

4.1. Data Inventory and Semantics

In industrial applications, availability of data is a concern. Data may be produced within old machinery or legacy system. In industrial manufacturing, a numerous range of heterogeneous data sources and IT systems of different types can be found, spanning various areas and processes along the enterprise. Commonly, machine data is collected from programmable logic controllers (PLC). However additional measurements can be added by retrofitting sensors to existing systems. Various control systems can be in place that facilitate acquisition of machine data, known as SCADA systems (supervisory control and data acquisition). A various range of protocols, e.g., OPC/UA or MQTT, are common in industrial automation and can be operated in parallel for several subsystems of one plant. Higher order systems also collect or provide relevant machine and process data, e.g., MES (manufacturing execution systems), ERP (enterprise resource planning), or CAQ systems (computer aided quality).

When planning a smart maintenance solution, it is necessary to get an overview of various data sources and systems within production, their availability and context. References to data sources can be made on different levels:

- Reference to availability within data infrastructure
- Reference to the machine function and operation
- Reference to the manufacturing and maintenance process.

The main purpose of a *data inventory* is to identify relevant data sources within the production and maintenance context. It is a prerequisite to integrate data sources and to make them available for smart maintenance solutions. A data inventory is

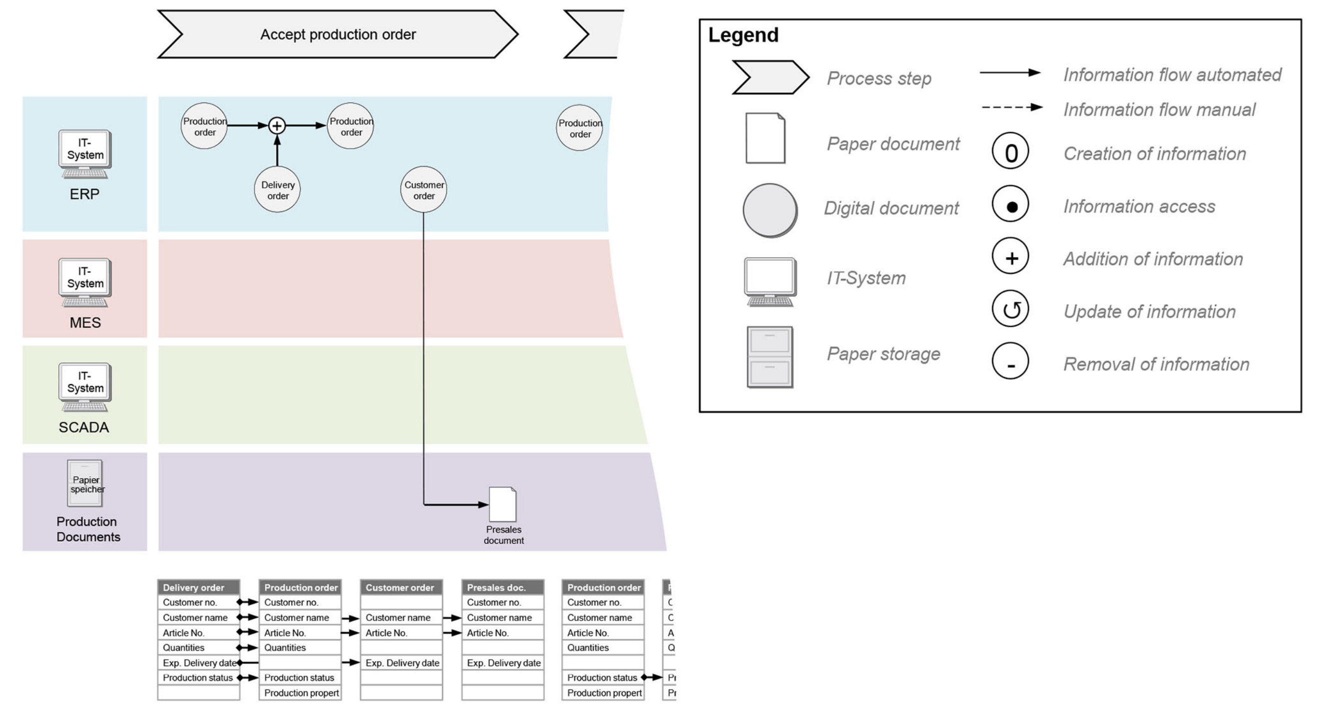


FIGURE 8 | Data Map for the visualization of data sources and data flow along the manufacturing process.

built by interviewing stakeholders relevant to producing data (e.g., automation experts, production planners, shift leaders, maintenance experts) and using existing documentation.

The *structure models* built within the domain understanding can be enhanced by adding references to data sources. In **Figure 4**, circles denote points of measurement at system elements. By back-tracing possible failures through the energy or information flow within the system (red arrows in **Figure 4**), signals can be found that are possible indicators for impeding failure (red circles in **Figure 4**). This gives data scientists important information about the data points and their relation to a machine function and operation and is also used for the design of models for smartmaintenance.

Within a *data map* as described by Joppen et al. (2019), the identified data sources are shown in context of the manufacturing and maintenance processes. The interconnection of data sources, as well as data flows between IT systems and production resources are given. The data map is based on the data inventory and gives context to the process information collected during domain understanding. It aids the project team and especially data scientists with the overall design of a smart maintenance solution and its embedding into the maintenance process from a use case point of view. We have enhanced the data map (Joppen et al., 2019) by improving the visibility of data flow in various data bases and IT systems. An example of our proposed enhanced data map is given in **Figure 8**, where the information is given in three lanes. The top lane shows a simplified view of production and maintenance process steps. In the middle lane, documents and databases are visualized that affect each process step. Documents and databases are structured by IT-Systems (colored rows). The bottom lane lists the data sources by their identifiers underneath the respective document or database. In the middle and lower lane, lines are used to visualize data flows.

The above-mentioned models for data inventory, structure models, and data map, are used for informal or semi-formal information gathering. They are informal means for *semantic data modeling*. The main intention is to enable the communication between different stakeholders in the manufacturing and maintenance domain, automation, IT systems, management and data analysts. The utilization of expert's knowledge is an important factor to implement efficient solutions, as it broadens the understanding of the analysts for the mechanical equipment of interest. These methods are assisting the knowledge transfer and, thus, the design of smart maintenance solutions from data and machine learning models to use case design.

Semantic data models also support the transfer to other plants. The equipment and the production process might differ between plants. However, the type of the machines used in the shop-floor in most of the cases is common, as the context of the production is the same (i.e., automotive parts). In addition to the informal and semi-formal models, formal semantic data models based on ontologies can be built. Especially in the case of common context, e.g., hot/cold-forming presses, scrap-belts, this semantic model can be a common dictionary of ontologies, offered to semantically describe each aspect of the production of each distinct plant, applying a uniform approach. Deploying smart maintenance solutions developed to process the data based on their semantics, increases the agility and the portability of the deployment, ameliorating multiple model transfer issues. Formal semantic models are also an important step toward automated setup and adaption of smart maintenance models, since they allow to overcome the problems of heterogeneous, unstructured data sources by giving context. However, the increased degree of formalization comes with an increased effort for building the respective semantic model. Hence, the semantic model and its degree of formalization (e.g., basic data inventory vs. a fully modeled ontology) should be decided based on cost and benefit considerations. Strategic arguments should be also considered, since a semantic model is also relevant in the context of building a digital twin of the production, which offers application scenarios and benefits.

4.2. Data Understanding

Data understanding is required in order to build detailed data models tailored to each specific use case. In BENTELER case, we have applied two monitoring approaches, a failure prediction and a fault detection one. For the current analysis, the difference between failures and faults is that, the former are serious equipment malfunctions that stop the production potentially for several hours, while the latter can cause minor deviations from the normal behavior for the equipment that usually affect the quality of the end product and might lead to a failure.

Both monitoring approaches are applied on preprocessed data in order to filter out noisy values usually encountered at the beginning or at the end of the production batches, or during idle periods of the equipment (e.g., applying maintenance actions, replacing specific equipment artifacts for the production of the next batch), or idle periods caused by bottlenecks in the production chain. A characteristic example is the scrap-belt use case, where the electrical current of the motors moving the belt is instantly increased in abnormal levels on every cold start and it drops to zero when the belt is not moving.

There are also monitoring policies that apply on the idle state of the equipment that consider the values obtained during the normal functionality as noise. An example is the monitoring of the hydraulic oil level of the hot-forming press use case. The oil tank of the hydraulic system is attached on the press, hence during the normal functionality of the press there are serious deviations in the oil level measurement due to the movement of the stamper of the press. In this specific use case we apply two policies, one that monitors the oil level at the highest pressure applied from the press (i.e., during the production of an item) and one that monitors the oil level when the press is idle.

4.3. Data Modeling

Each distinct monitoring policy uses a different data model to identify when the press is moving or not. The failure prediction monitoring approach uses a motif detection algorithm (Yeh et al., 2017), to map sets of sensor measurements to artificial events, in order to apply algorithms for event-based prediction inspired by the aviation industry, like the one proposed in Korvesis et al. (2018). The reasoning behind the measurement to event mapping is that, before a major failure in the equipment, there might be indications in the form of repeating events (e.g., minor faults, anomaly behavior), which, if be identified early, can potentially predict the upcoming failure. These repeating events along with historical information regarding major failures, are used to train Random Forests models for failure prediction.

The fault detection approach is used to complement the prediction approach as it can be used on cold start with some basic parameterization. The detection is based on an ensemble of unsupervised monitoring approaches. As the nature of the faults (i.e., the sensor measurements footprint) might differ between different fault types, a single monitoring approach would not be enough to cover all the cases. For example, a fault might have a footprint with spikes on the measurements, hence an approach that monitors the trend of the measurements, would not be able to detect it. Our proposed fault detection approach uses (i) a distance-based outlier detection algorithm on streaming data (Georgiadis et al., 2013), to detect abnormal values (including spikes) in the measurements, (ii) a linear regression algorithm combined with lower-upper thresholds for detection based on the trend of the measurements and (iii) a simple threshold-based approach as a fail-safe mechanism if all the other approaches fail to identify a fault.

4.4. Best Practices

The semantic representation of each aspect of the production enhances the uniformity toward the model transfer goal. When deciding for the needed degree of formalization for a semantic representation, cost-benefit and strategic considerations should be considered. In the example of BENTELER, graphic system structured models have been selected in lieu of full-fledged ontology models, since it followed a more hands-on approach and straight exploitation in the design of a smart maintenance solution.

The understanding of the data based on the obtained domain knowledge is important, in order to identify the relevant data sources and to select, filter, transform and combine the most appropriate features in each use case. The latter preprocessing steps need to take place in isolation from the actual data processing for fault detection and failure prediction, in order to provide context agnostic solutions toward the utilization of the full spectrum of the capabilities that the semantic representation can offer.

5. PROCESS INTEGRATION AND TRANSFER

5.1. Process Integration

The data infrastructure presented in section 3, offers a containerized applications management system for the automation of the deployment and the handling of the scaling in the BENTELER Developers Space. It offers a web interface, on which developers can upload configuration files for the deployment of the docker containers. The same configuration files are used to deploy multiple containers, supporting different instances of the same micro-services, upon request, for scenarios where the availability of the service is crucial or when the incoming throughput of sensorial measurements is above the capacity of a single container. Developers also provide configuration files for deployment using the docker-compose

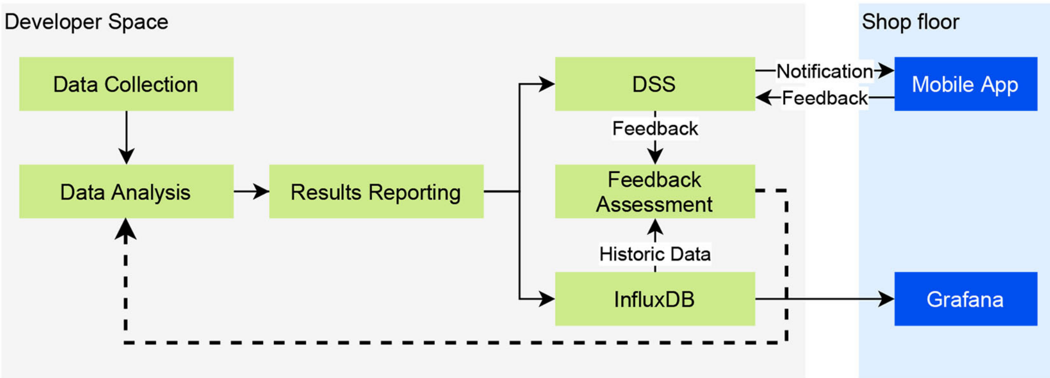


FIGURE 9 | Process integration scenario.

tool outside the provided container management environment of the Developers Space, in order to support the deployment on BENTELER plants that might not communicate with the Developers Space or they do not have a local Developers Space infrastructure.

The platform offers a batch run feature, where configuration files are uploaded, containing information regarding the tasks that need to be started and their configuration. A single file can be constructed by the data scientists, either per use case, or per BENTELER plant containing multiple use cases, which includes the required information to deploy the tasks in the platform with minimum effort. For the monitoring and handling of the running tasks, the platform offers a web interface, where the user can start, stop or get details (i.e., parameterization) regarding each running task.

Figure 9 presents a process integration scenario depicting the communication between the different components in order to obtain the data, analyze them, provide the results to the maintenance engineers and receive their feedback. The figure is divided into two areas, the Developers Space and the Shop floor. We consider that the developers and the data scientists act in the Developers Space, while the maintenance engineers in the Shop floor interacting with the respective components of the system.

The first step of the process is the Data Collection, where appropriate bridges are built to transfer the data from the Shop floor to the Data Analysis components. The data are analyzed, and the results are sent to the Results Reporting component, which is responsible for the circulation of the results. In the presented scenario the results are stored in a time series database and sent to a Decision Support System (DSS).

The smart maintenance platform encapsulates a Grafana platform[1] for the visualization of the results. Grafana uses the time series database of the Developers Space as a data source, hence it can visualize both the analysis results and the persisted sensorial measurements. Multiple Grafana dashboards are provided visualizing valuable information for both the data scientists and the maintenance engineers. The data scientists can visually assess the sensitivity of the different parameterizations of the data analysis tasks, inspecting the number of reports per task, in combination with information from the maintenance logs. The maintenance engineers can set automated rules in Grafana, in order to get alerted inside the Grafana platform when a plotted measurement satisfies specific criteria, e.g., when two or more detection monitoring tasks with different parameterizations report that something is going wrong on the same sensorial input.

Apart from the Grafana dashboards, maintenance engineers are equipped with mobile devices with a notification's application installed, which directly communicates with the DSS. Whenever a data analysis result arrives in the DSS, a set of pre-defined rules is activated, in order to apply an initial assessment of the situation and, if deemed necessary, to notify the appropriate maintenance engineers. The engineers apply the final assessment and the required actions based on the provided information. A feedback mechanism is also deployed, which allows the engineers to evaluate the analysis results providing a rating and a free text comment. The rating information can be directly used for the retraining of the failure prediction models or the reconfiguration of the fault detection tasks. The provided comments can be manually processed by the data scientist in order to extract knowledge.

5.2. Transfer of Models

The aforementioned phases from domain understanding to data modeling and process integration have been described in context of a single production line in one plant. However, the solution can be transferred to similar production lines, as well as other plants, thus reducing the overall development cost for one deployed smart maintenance solution. A successful transfer requires the transfer of detection or prediction models and must consider the underlying data models, infrastructure, as well as the specifics of the use case, i.e., details about machines together with the specific production and maintenance processes at the plant under consideration.

A *solution pattern* combines all necessary technical components, as well as supporting documentation for

[1]https://grafana.com/

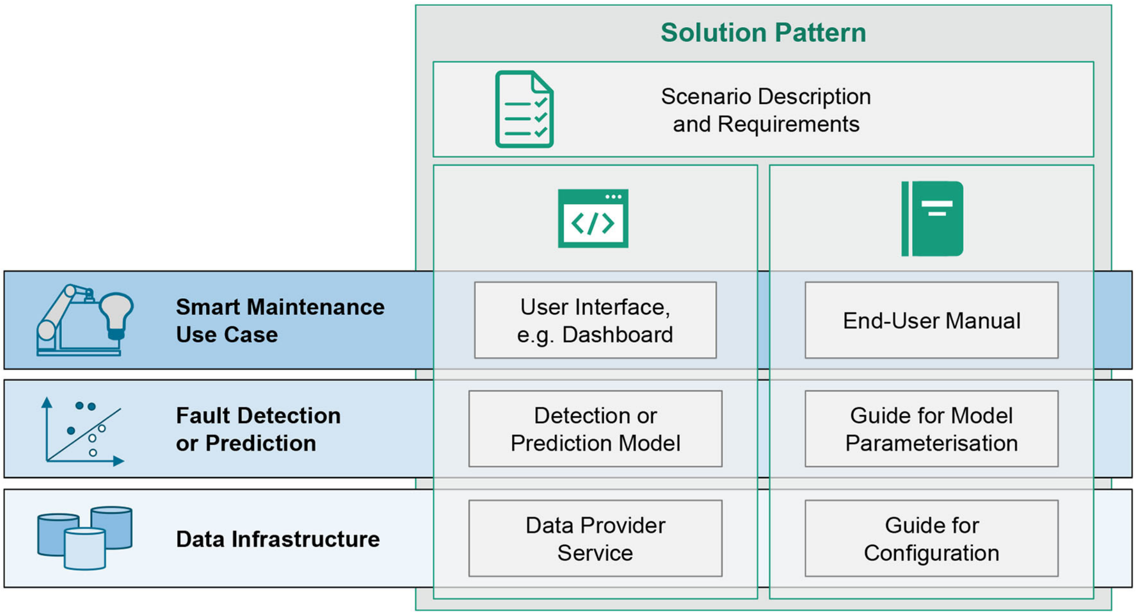


FIGURE 10 | Components of a smart maintenance solution pattern.

deployment and application of the solution. As a reusable pattern, it facilitates the adaption and deployment on new infrastructure. It thus covers the following items of a smart maintenance solution:

- Program code or container of the implemented detection/prediction model,
- Program code or container for the visualization or dashboard,
- End-user manual for the deployed dashboard or visualization and its application within the production and maintenance process, e.g., how to handle alerts,
- Guide to parameterization of the model according to specific machines and sensors, e.g., how to set thresholds,
- Guide to technical configuration according to specific data infrastructure, e.g., local service architecture, data base schemes.

A solution pattern is preceded by a scenario description. It is a short briefing describing the scope and the requirements to the solution and allows a quick selection and decision for a fitting solution to a problem at hand. In **Figure 10**, the technical components, as well as documentation components constituting a solution pattern are summarized. It is important that guides and manuals are kept concise and consistent between various solutions. This reduces documentation effort during development and implementation effort during deployment at the same time.

The modular description of the solution pattern ensures a clear distinction between general-purpose parts of a solution and individual parts. Even if use case specific implementation may be necessary for an individual smart maintenance solution, the modular description still allows the efficient reuse of tested components of solution patterns. Overall, using pre-tested and evaluated components not only allows the transfer of functionality, but it also increases acceptance of smart maintenance among different plants and users.

In a next maturation level, an automated configuration and parameterization can be developed based on the documentation. A good data quality and the availability of formal and semantic data models based on a common vocabulary are required, in order to allow a profound automation. Automated solutions simplify the application in other plants and push toward a commoditization and servitization of smart maintenance, e.g., in the form of a smart maintenance app store.

However, most current real-world scenarios, even within a corporation like BENTELER, are characterized by individual specificities and preconditions. The development effort for unified and standardized infrastructure and/or common semantic models needs to be justified, to proceed to such an investment. The aforementioned solution patterns thus offer a good balance between individual adaption efforts and general-purpose development efforts. They are a prerequisite to fully automated deployment, and thus a practical and useful solution for most current use cases.

5.3. Best Practices

Containerization solutions facilitate the deployment of the solutions enhancing ITS transfer capability to other plants. Maintenance engineers should focus on the results of the data analysis and not on the parameterization and the configuration of the analysis. Hence, it is important to automate to the possible extend the configuration or provide a set of pre-configured tasks. Clean and user-friendly interfaces should be provided for the results presentation and visualizations provide a safe option as they clearly depict the issue and allow its tracing

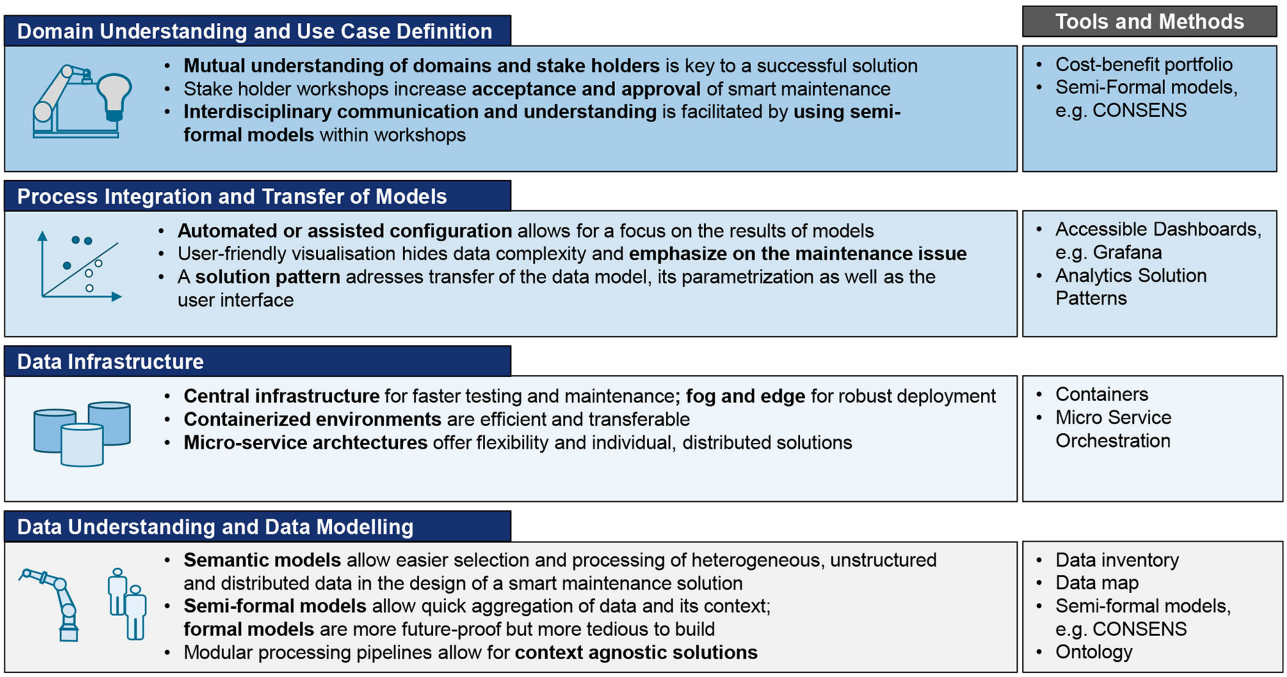


FIGURE 11 | Summary of best practices for successful implementation and deployment of a smart maintenance solution in industrial manufacturing.

through the plotting of historical information. A mobile device-based notifications mechanism is also useful, to make sure that the engineers are informed on time for the detected or predicted faults.

The utilization of solution patterns, facilitates the adaption and deployment of the PdM solution on new infrastructure, by reducing the documentation and implementation effort during the development and deployment phases, respectively.

6. EMPIRICAL RESULTS AND BEST PRACTICES

This section, provides an outline of the best practices and the proposed tools and methods, presented in the previous sections. As **Figure 11** depicts, the communication between the domain experts, the data scientists and the developers, is important in order to achieve the knowledge transfer for the domain understanding and the use case definition. The process can be supported by semi-formalized methods.

In the process integration phase, the focus should be on the maintenance issues and the results of the models. Automated or assisted configuration of the software solution and user-friendly interfaces are proposed, to avoid distracting the maintenance experts' attention from the actual maintenance issue and the guidelines for its mitigation. The adoption of a solution pattern aids the transfer of the software implementation to similar use cases providing a holistic PdM solution to the company.

The data infrastructure contributes significantly to the implementation and model transfer phases. A centralized infrastructure offering a containerized environment eases the development of a micro-service based PdM solution offering flexibility and easy distribution of the workload.

The context awareness is important for the domain and data understanding, which in turn are important for the data modeling. However, the end solution should be context agnostic in order to be easily transferable to a wider range of use cases. The utilization of semantic models and semi-formalized methods strengthen the agnostic nature of the PdM solution.

For the scrap belt use case, a fault detector could be developed rapidly using the presented methodology. Already within the observed period of roughly 1 year, four alarms have been given for respective incidents by the fault detector. The mean time to repair (MTTR) has thus been reduced from 6 to 4 h, and the mean time between failure (MTBF) has increased from 30 to 180 days. With further improvement of the fault detector, a higher sensitivity or even prediction is expected to increase the number of predicted incidents to 11 per year, thus resulting in an expected further reduction of MTTR to 2 h and increase of MTBF to 365 days.

Fault detection and maintenance dashboards have been transferred to multiple sites, thus leveraging the benefit for BENTELER even further.

7. CONCLUSION

This work highlights the crucial aspects of the PdM implementation process, toward the integration of a smart maintenance solution within a company. Through practical examples, which are derived from a lighthouse factory (i.e., BENTELER plant) within the BOOST 4.0 project, a business-driven process for the implementation of predictive analytics is proposed.

The process is divided into three main phases Use Case Analysis, Proof of Concept, and Deployment, while four main aspects need to be considered in each phase: Analytics Use Case, Data Sources, Data Infrastructure and Data Analysis. The first process phase, Use Case Analysis, includes the domain understanding and the use case analytics sub-phases, which enable the Use Case definition. The Proof of Concept phase is comprised by the data acquisition, data fusion and data pre-processing sub-phases, which facilitate the data understanding and modeling. The final phase, i.e., Deployment, includes the process integration, the roll-out and the scale-up sub-phases, providing a holistic solution to the company.

In order to effectively address the PdM implementation process, the work presents a set of best-practices, proposed tools and methods for each one of the Domain Understanding and Use Case Definition, Data infrastructure, Data Understanding and Modeling and Process Integration and Transfer of Models sub-processes.

The proposed methodology has been successfully applied to multiple BENTELER plants, leading to reduced mean time to repair and significantly increased mean time between failure.

DATA AVAILABILITY STATEMENT

The data analyzed in this study is subject to the following licenses/restrictions: Production data from BENTELER Automotive subject to non-disclosure. Requests to access these datasets should be directed to Sebastian von Enzberg, sebastian.von.enzberg@iem.fraunhofer.de.

AUTHOR CONTRIBUTIONS

DK and IM contributed to the domain understanding, use case definition, and analysis. AN contributed to the data infrastructure, data modeling, and process integration. SE and AK contributed to the overall framework of the development process, methods for use case definition, data inventory, and model transfer. All authors contributed to the best practice collection. All authors contributed to the manuscript revision, read, and approved the submitted version.

FUNDING

This research work is funded by the BOOST4.0 project funded by European Union's Horizon 2020 research and innovation program under grant agreement No. 780732.

REFERENCES

Bokrantz, J., Skoogh, A., Berlin, C., Wuest, T., and Stahre, J. (2020a). Smart maintenance: a research agenda for industrial maintenance management. *Int. J. Product. Econ.* 224:107547. doi: 10.1016/j.ijpe.2019.107547

Bokrantz, J., Skoogh, A., Berlin, C., Wuest, T., and Stahre, J. (2020b). Smart maintenance: an empirically grounded conceptualization. *Int. J. Product. Econ.* 223:107534. doi: 10.1016/j.ijpe.2019.107534

Bumblauskas, D., Gemmill, D., Igou, A., and Anzengruber, J. (2017). Smart maintenance decision support systems (SMDSS) based on corporate big data analytics. *Expert Syst. Appl.* 90, 303–317. doi: 10.1016/j.eswa.2017.08.025

Gausemeier, J., Frank, U., Donoth, J., and Kahl, S. (2009). Specification technique for the description of self-optimizing mechatronic systems. *Res. Eng. Design* 20:201. doi: 10.1007/s00163-008-0058-x

Georgiadis, D., Kontaki, M., Gounaris, A., Papadopoulos, A. N., Tsichlas, K., and Manolopoulos, Y. (2013). "Continuous outlier detection in data streams: an extensible framework and state-of-the-art algorithms," in *Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data*, (New York, NY), 1061–1064.

Joppen, R., Enzberg, S., Kühn, A., and Dumitrescu, R. (2019). "Data map–method for the specification of data flows within production," in *Procedia CIRP* (Golf of Naples: Elsevier), 461–465.

Korvesis, P., Besseau, S., and Vazirgiannis, M. (2018). "Predictive maintenance in aviation: Failure prediction from post-flight reports," in *2018 IEEE 34th International Conference on Data Engineering (ICDE)* (Paris: IEEE), 1414–1422.

Kühn, A., Joppen, R., Reinhart, F., Röltgen, D., von Enzberg, S., and Dumitrescu, R. (2018). "Analytics canvas–a framework for the design and specification of data analytics projects," in *Procedia CIRP* (Nantes: Elsevier), 162–167.

Moens, P., Bracke, V., Soete, C., Vanden Hautte, S., Nieves Avendano, D., Ooijevaar, T., et al. (2020). Scalable fleet monitoring and visualization for smart machine maintenance and industrial IoT applications. *Sensors* 20:4308. doi: 10.3390/s20154308

Reinhart, F. (2016). Industrial data science–data science in der industriellen anwendung. *Industrie 4.0 Management* 32, 27–30.

Shearer, C. (2000). The CRISP-DM model: the new blueprint for data mining. *J. Data Warehous.* 5, 13–22.

Uhlmann, E., Franke, D., and Hohwieler, E. (2019). Smart maintenance–dynamic model-based instructions for service operations. *Proc. CIRP* 81, 1417–1422. doi: 10.1016/j.procir.2019.04.327

Yeh, C.-C. M., Kavantzas, N., and Keogh, E. (2017). Matrix profile IV: using weakly labeled time series to predict outcomes. *Proc. VLDB Endowm.* 10, 1802–1812. doi: 10.14778/3137765.3137784

Zheng, P., Sang, Z., Zhong, R. Y., Liu, Y., Liu, C., Mubarok, K., et al. (2018). Smart manufacturing systems for industry 4.0: conceptual framework, scenarios, and future perspectives. *Front. Mech. Eng.* 13, 137–150. doi: 10.1007/s11465-018-0499-5

Conflict of Interest: AN and IM were employed by company ATLANTIS Engineering SA. DK was employed by BENTELER Automobiltechnik GmbH.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

frontiers in Mechanical Engineering

ORIGINAL RESEARCH
published: 17 December 2020
doi: 10.3389/fmech.2020.578379

Intelligent Predictive Maintenance and Remote Monitoring Framework for Industrial Equipment Based on Mixed Reality

Dimitris Mourtzis, John Angelopoulos and Nikos Panopoulos*

Laboratory for Manufacturing Systems and Automation, Department of Mechanical Engineering and Aeronautics, University of Patras, Patras, Greece

The currently applied maintenance strategies, including Reactive and Preventive maintenance can be considered obsolete. The constant improvements in Information and Communication Technologies as well as in Digital Technologies along with the increase of computational power, have facilitated the development of new Artificial Intelligence algorithms to integrate cognition in computational systems. This trend is posing a great challenge for engineers, as such developments will enable the creation of robust systems that can monitor the current status of the machines and by extension to predict unforeseeable situations. Furthermore, Smart Computers will be capable of examining all possible scenarios and suggest viable solutions in a fraction of time compared to humans. Therefore, in this paper, the modelling, design and development of a Predictive Maintenance and Remote Monitoring system are proposed, based on the utilization of Artificial Intelligence algorithms for data acquisition, fusion, and post-processing. In addition to that, the proposed framework will integrate a Mixed Reality application for the intuitive visualization of the data, that will ultimately facilitate production and maintenance engineers to monitor the condition of the machines, and most importantly to get an accurate prediction of the oncoming failures.

Keywords: artificial intelligence, predictive maintenance, remote monitoring, augmented reality, machine learning

OPEN ACCESS

Edited by:
Dimitris Kiritsis,
École Polytechnique Fédérale de Lausanne, Switzerland

Reviewed by:
Somnath Chattopadhyaya,
Indian Institute of Technology Dhanbad, India
Marco Macchi,
Politecnico di Milano, Italy

****Correspondence:***
Dimitris Mourtzis
mourtzis@lms.mech.upatras.gr

Specialty section:
This article was submitted to Digital Manufacturing, a section of the journal Frontiers in Mechanical Engineering

Received: *30 June 2020*
Accepted: *30 October 2020*
Published: *17 December 2020*

Citation:
Mourtzis D, Angelopoulos J and Panopoulos N (2020) Intelligent Predictive Maintenance and Remote Monitoring Framework for Industrial Equipment Based on Mixed Reality. Front. Mech. Eng. 6:578379. doi: 10.3389/fmech.2020.578379

INTRODUCTION

Maintenance of industrial equipment as a part of the manufacturing lifecycle, approaches 60–70% of the overall cost of production. Therefore, being able to predict and perform machine maintenance operations in a short period of time can lead to successful troubleshooting, and at the same time increase the availability of machine tools (Mourtzis et al., 2015). Currently, inadequate maintenance techniques can reduce the total productive ability of the plant by between 5 and 20% (Wollenhaupt, 2016). Traditionally, maintenance professionals have combined several techniques, both quantitative and qualitative, with the aim to anticipate potential problems and alleviating downtime in their production plants. Predictive maintenance gives them the potential to optimize maintenance tasks in real time, maximizing the useful life of their equipment while avoiding disruption of operation. Recent studies also show that unplanned downtime is costly, with an estimation of $50 billion per year for global producers (Deloitte, 2017a). In the Industry 4.0 environment, maintenance should do much more than simply prevent the downtime of individual assets. Predictive maintenance increases uptime by 10–20%, while reducing overall maintenance costs by 5–10% and maintenance planning time by 20–50%. Furthermore, due to increased interconnectivity and new opportunities for collecting, processing, and analyzing information, predictive maintenance can be a very powerful strategy (Deloitte, 2017b).

In addition to failure prediction, a significant challenge is the implementation of reliable and error-free maintenance operations and, as a result, the constant validation of fully working equipment as soon as possible. To that end, a significant amount of development work has been made to design and improve real-time technical service systems and software focused on mobile apps to prevent unwanted errors and malfunctions (Masoni et al., 2017). Moreover, the handling of complex cases of smart factories, intelligent maintenance, self-organized adaptive logistics, customer-integrated engineering and smart factory architectures require the integration of production data into modeling that can only be achieved with the use of advanced simulation and Information Technology (IT) (Mourtzis, 2020). Moreover, the value of products will eventually be focused on their software parts not on their specification or implementation functions under the Product Service Systems (PSS) paradigm (Mourtzis et al., 2018). With the advancement of Information and Communication Technology (ICT) and cutting-edge technologies such as Mixed-Reality (MR), Augmented Reality (AR) and Virtual Reality (VR), the academic domain is expanding this strategy by leveraging the advantages of AR for data visualization during maintenance operations (Mourtzis et al., 2017; Mourtzis et al., 2018; Palmarini et al., 2018). Emerging technologies such as the Internet of Things (IoT), cyber-physical networks and cloud computing have enabled the processing of vast volumes of tracking data, which is intended to significantly increase manufacturing productivity (Tao et al., 2018; Fantini et al., 2020). However, as a core topic in prognostics and health management, the remaining useful life (RUL) prediction based on monitoring data ca be used to prevent a failure triggered (Lei et al., 2018). RUL prediction is thus a hot topic that has drawn more and more interest in recent years (Yang et al., 2019). Most research studies on intelligent prognosis and health management (PHM) analysis using data-driven approaches by deducing correlations between data from different sensors (e.g. accelerometers, acoustic energy emission) to determine the remaining useful life (e.g. accelerometers, acoustic energy sounders, etc.). In order to limit the complexity inherent in the dynamic updating of online data, Machine Learning has arisen as a way of analyzing vast volumes of data for statistical purposes. Especially in the implementation of neural network-based techniques, complex multidimensional non-linearities can be used for automated learning, allowing for efficient processing of data features in an attempt to provide optimized solutions (Vogl et al., 2019).

Having identified the above-mentioned challenges, this research work presents the design and development of a predictive maintenance framework for industrial equipment. Further to that the contribution of this research work extends to the presentation of a custom Data Acquisition (DAQ) device and a framework for processing the data via the Digital Twin of the equipment for the calculation of Remaining Useful Life of critical components. The remainder of the paper is structured as follows. In *State of the Art* the most pertinent literature is reviewed, and commercial devices are compared. In *Proposed System Architecture*, the proposed system architecture is presented. In *System Implementation* the practical implementation steps are discussed. Then in *Case Study*, an experimental case study that has derived from Industry is presented and the results are discussed. Finally, in *Concluding Remarks and Outlook*, the paper is concluded, and future research points are discussed by the authors.

STATE OF THE ART

Machine Learning

Among the latest trends in the modern manufacturing world, is the so-called AI. An also well-known subset of the above-mentioned concept is Machine Learning (ML). Concretely ML algorithms are defined as computer-based algorithms that improve their efficiency through experience, i.e. data processing (Mitchell, 1995). Globally the AI adoption is surging at enormous rates, as it becomes apparent in the report presented in (Hupfer, 2019), from where it can be concluded that AI adoption marked a surprising 270% increase in a timespan of 4 years along with an increase in global spending of around 80 billion dollars **Figure 1**.

Additionally, Deep Learning (DL) techniques have been applied for the integration of systems in edge computing, setting edge nodes in edge services and terminal devices, using DL architectures for predictive analysis with quick preprocessing and accurate performance classification to assess the life expectancy of components. A classification of the most common DL frameworks is as follows:

- Neural Networks (NN) (Chryssolouris, 2006; Chen et al., 2019)
- Deep Neural Networks (DNN) (Zhao et al., 2017)
- Convolutional Neural Networks (CNN) (Li et al., 2018; Mourtzis et al., 2020a)
- Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) (Zhao et al., 2017)
- Gated Recurrent Units (GRU) (Chen et al., 2019)
- Recurrent Neural Network (CNN-RNN) (Banerjee et al., 2019)
- Convolutional Neural Network and Long Short-Term Memory (CNN-LSTM) (Kong et al., 2019)
- Convolutional Neural Network and Gated Recurrent Unit (CNN-GRU) (Lei et al., 2018)

As Industry 4.0 continues to evolve, many companies are struggling with the realities of AI implementation. Indeed, the benefits of PdM such as helping determine the condition of equipment and predicting when maintenance should be performed, are extremely strategic. The implementation of ML frameworks can lead to major cost savings, higher predictability, and increased availability of the systems. Therefore engineers have focused their efforts on the development of new technologies and techniques for facilitating the prediction of manufacturing equipment malfunctions and therefore to further optimize

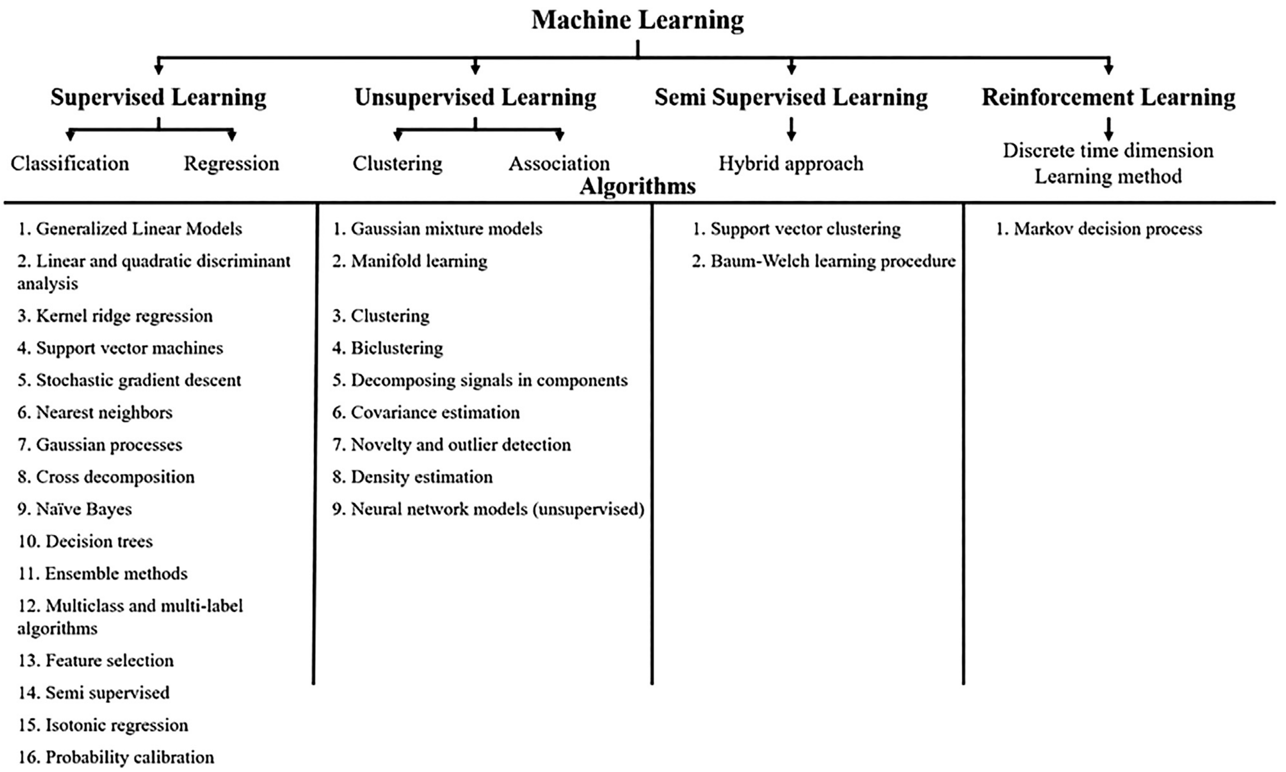


FIGURE 1 | Machine Learning Types & algorithms (Adapted from Ahmed and Khan, 2019).

existing maintenance policies as well as to introduce more adaptive maintenance policies. PdM can be defined as a series of processes, where data is collected over time in order to monitor the state of equipment, in a manufacturing system. Ultimately, the goal is to identify/recognize patterns that in turn will facilitate engineers to predict and ultimately prevent failures (Rezaeianjouybari and Shang, 2020). Some of the most common problems that can be addressed with PdM include, the calculation of Remaining Useful Life (RUL), which aims at the suitable scheduling of future Maintenance and Repair Operations (MRO), Flagging Irregular Behavior, which is based on anomaly detection by the utilization of time series analysis, and Failure Diagnosis and Recommendation of Mitigation after failure (Lei et al., 2018; Mourtzis et al., 2020a). While certain Facility Managers perform PdM, this has been done traditionally by using Supervisory Control and Data Acquisition (SCADA) systems set up with human-coded/hard-coded thresholds, alert rules, and configurations. However, this is a semi-manual approach that does not take into consideration the more complex dynamic behavioral patterns of the machinery, or the contextual data relating to the manufacturing process, thus lacking adaptability relative to the current status of the industrial equipment (Nicholson et al., 2012). What is more, in recent research works SCADA systems are integrated with ML algorithms, in order to extend their usability as well as to shift towards prognostics (Pang et al., 2020; Ruiming et al., 2020; Zhang and Lang, 2020). In the research work of Wang et al. (2020), the authors have developed a framework based on Convolution Auto Encoder and Long-Short Term Memory (LSTM) in an attempt to estimate RUL more accurately in comparison to conventional methods. For the recognition of patterns, which facilitates the process of building the predictive model, data exploration techniques must be utilized so that the engineer can determine whether the data includes degradation or failure patterns (Erfani et al., 2016; Li et al., 2019).

Remaining Useful Life

As the name indicates, Remaining Useful Life, also referred to as RUL, describes a wide variety of algorithms which aim to predict the remaining life of assets and/or their components, ultimately developed under a predictive maintenance framework. According to Baru (2018) there can be identified three basic techniques regarding the calculation of RUL based on the data that are available, namely lifetime data, run-to-failure data, and known threshold data. An interesting approach in presented in (Loutas et al., 2013) for the calculation of RUL for rolling bearings based on the utilization of ε-Support Vector Regression (ε-SVR), concluding that linear models cannot provide accurate results since there is non-linear between the features extracted by the data spectral analysis and the RUL prediction. Another aspect of the usefulness of RUL estimation is presented by (Sun et al., 2020). The authors have implemented a framework for the RUL calculation of cutting tools, thus managing to increase the environmental sustainability of the cutting tools by 8.39% per flute. From the investigation of the available literature it can be concluded that the estimation of RUL is a challenging topic, requiring exhaustive data processing. Further to that in the majority of the publications it is implied that linear approximations regarding the degradation of the physical system are not sufficient in terms of accuracy, as presented in (Yang et al., 2021), where the authors investigated the prediction of RUL in induction motors. The authors in (Wen et al., 2021) has proposed a generalized methodology for the prediction of RUL based on the fusion of multiple signals. It is worth noting that they achieved an increase in terms of accuracy of approximately 10 percent. Kozjek et al. (2020) have also presented an interesting research work on the prediction of RUL with the utilization reinforcement learning, which is compared with two other algorithms, indicating promising results. In the research work of (Liu et al., 2019), a RUL prediction framework is proposed based on Health Index comparison, making it suitable for cases where there is limited amount historical data. It is stressed out that the topic of RUL prediction is still challenging for engineers and by extension there is plenty of room for improvement. In addition to that, the use of Digital Twin could compensate the lack of raw data from machines, with the generation of fault datasets.

Extended Reality

Among the latest developments of the current industrial revolution, advances in high-end digital technologies are entailed, including Extended Reality (XR). In its essence, XR is an umbrella term, often used by engineers and researchers around the world, in order to describe technologies such as Augmented Reality (AR), Mixed Reality (MR), and Virtual Reality (VR) (Mourtzis et al., 2020b). The two former technologies are very close, since they are based on the partial immersion of the user to a virtual environment, while the latter, implies the total user immersion in a virtual, computer generated environment. In addition to that, what differentiates AR from MR is the fact that MR is based on the user interaction with the digital information, also known as holograms in that case (Fast-Berglund et al., 2018). The use of AR in maintenance is an important aspect that has to be further researched under the Industry 4.0 framework. Since new technologies are constantly becoming available, existing techniques could be leveraged so as to increase the efficiency of maintenance tasks, minimize the errors and the risks imposed in such operations. As presented in the research work of Vorraber et al. (2020), both maintenance technicians and experts are keen on integrating AR and MR solutions in their line of job, in order to achieve better communication and most importantly to limit the complexity of the maintenance procedures. Although the maturity level of AR applications has increased during the last decades (Mourtzis et al., 2020c), there are constantly arising new challenges, such as the integration of Predictive Maintenance and AR/MR so that digitalization of the manufacturing processes becomes a reality (Wolfartsberger et al., 2020). Further to that in two recent systematic literature reviews, presented by Palmarini et al. (2018) and Egger and Masood (2020) the current implementations of AR are based on manual solutions and the use of Predictive Analytics/Prognostics has not been yet faced, thus indicating that there is fertile ground for further research in that field.

PROPOSED SYSTEM ARCHITECTURE

In the following paragraphs the proposed system architecture will be discussed in detail. The key aspects of the proposed methodology are the DAQ device, which conforms to the latest IoT standards. However, in order to efficiently monitor the status of professional refrigeration systems, they have to be analyzed into two subsystems, namely the cooling chamber of the refrigerator and the compressor compartment. These two subsystems often are not located in the same room/building, thus require different DAQ devices to be installed. By the virtue of the diversity of installed sensors, crosschecking the measured values, is enabled and therefore more accurate predictive models can be trained. The general architecture of the proposed is depicted in **Figure 2**.

Data Acquisition Device

In this section the architecture of the framework for the DAQ device will be discussed. For the DAQ module, two main aspects will have to be investigated, namely the DAQ device and the communication interfaces as well. The development of the DAQ device is based on the design of a custom circuit board in combination with an Arduino micro-controller which incorporates all the required modules for the data acquisition from the sensors attached to the board, the pre-processing of the data, an interface for user interaction and a wireless network module for the data transmission to the Cloud Database. In order to make the DAQ device adaptive to the customer needs, and subsequently to support a wide variety of configurations, the sensor modules are not hardwired/soldered on the main PCB of the DAQ device. More specifically, the PCB supports wired connectivity, through 3.5 mm jack ports for the sensors. However, in order to enable the communication between the DAQ device (**Fig. 3A**) and the Cloud Database an RF-based Wireless Sensor Network (WSN) is utilized. For the setup of the WSN, XBee modules are utilized. More specifically, an XBee module (**Figure 3B**) is installed on each of the DAQ devices and another one is installed on a computer which acts as the network coordinator. For the correct communication of the DAQ devices to the computer, each RF module is tagged. Furthermore, during the data transmission, the data packets are also including the tag of each RF tag, so that the received data can be correlated to the corresponding machine.

As far as the sampling rate is concerned, the DAQ device collects feedback from the installed sensors on a varying rate. The sampling rate for the accelerometer sensor is set to one (1) second or 1 Hz. As far as the sampling rate for the temperature sensor is set to 5 s and for the pressure sensor is set to 10 ms (milli-second) as soon as a surge event is detected. However, if the customers require a different resolution regarding the data collection and by extension the estimations made by the framework, then they

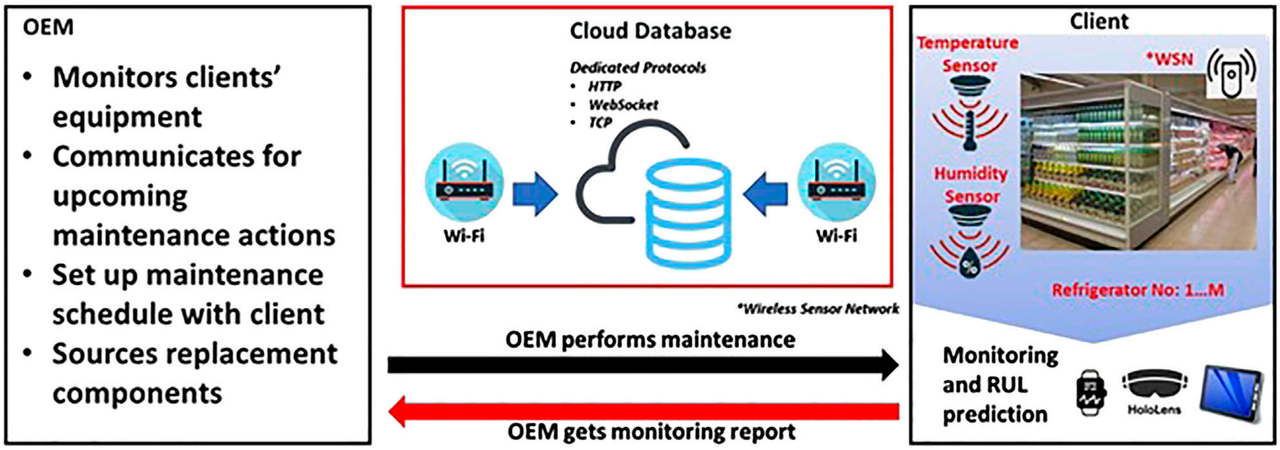


FIGURE 2 | Proposed System Architecture.

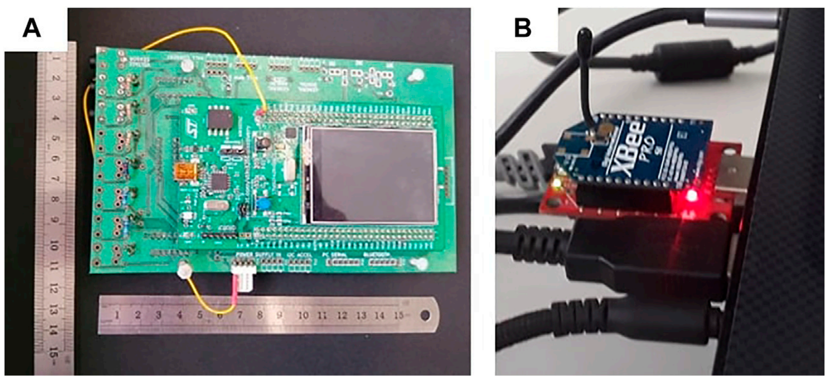


FIGURE 3 | Prototype DAQ device.

adjust the sampling rates. In **Figure 3**, the prototype board for the DAQ device is presented.

Digital Twin Development

In order to create a suitable framework for the predictive maintenance functionality of the equipment, the design and development of a Digital Twin is required. The aim is to analyze the data gathered from the DAQ devices and based on the simulation model to predict future equipment malfunctions. Therefore, in the case of the refrigerators, the physical model is created in the Simulink programming environment. For the simulation of the model, MATLAB is also utilized for handling the imported data as well as setting up the simulation parameters. The physical model of the refrigerator is fully parametrically designed so that it can be adapted to the technical specifications of the physical system. In **Figure 4**, the developed model within the Simulink environment is presented.

It is stressed out that the model consists of several subsystems, or else functional blocks, in an attempt to increase the resolution of the simulation model, such as the compressor, the evaporator, the condenser, and the refrigerator compartment. In the "Data Input" block, the data from the DAQ device are imported to the model. Then the standard refrigeration cycle for refrigeration is run and the results are plotted. Through the plots, crucial parameters of the refrigerator, such as temperature, power consumption and pressures within the refrigerant distribution network can be observed. For the simulation, the fluid properties of the R134a refrigerant were also imported in the model. As a result with the proposed methodology, it is possible to predict future asset malfunctions based on the simulation of the refrigeration cycle and plan accordingly their production schedule so that the equipment downtime is further minimized. In addition to that, the simulation results are also combined/fused with the data gathered from the physical machine so as to predict the RUL of specific components of the equipment. The usability of the Simulink model extends also to the generation of fault data. For instance, in the refrigerators a common failure is the loss of pressure in the refrigerant distribution network. Therefore, in the existing model the "fault" is simulated with the addition of an array of blocks,

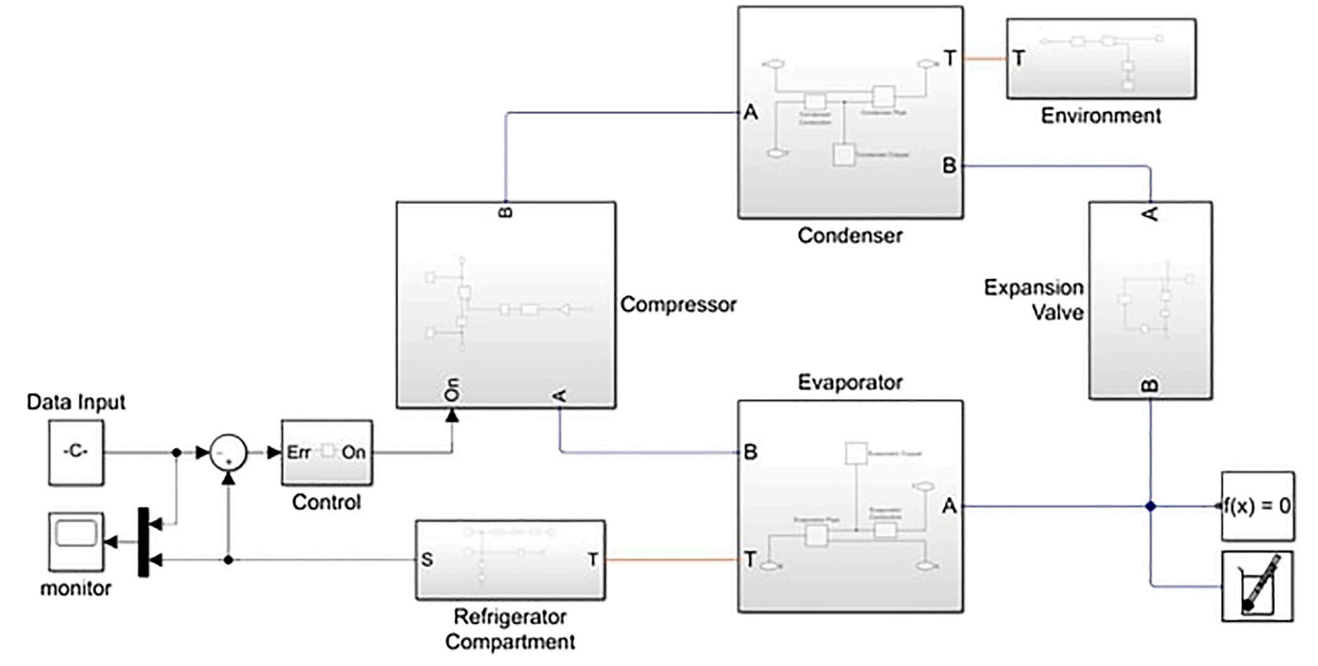


FIGURE 4 | Refrigerator model in MATLAB Simulink.

based on which the differential pressure in specific subsystems, such as the compressor pressure differential is offset to a fault value. Then after the corresponding datasets for the healthy state and the fault state have been generated, the model automatically recalls the RUL algorithm.

In its essence the RUL algorithm utilizes data from both the digital twin and the physical model for the prediction of the time, in hours, before maintenance is required. The first step in the RUL algorithm is the Fourier transformation of the signals to the frequency domain. The next step is the creation of the spectrograms for each of the under-examination parameters, e.g. pressure inlet and outlet in the compressor, vibration signal from rotational components. In this step, two spectrograms are created, one for the fault data and one for the healthy data. Based on the spectrograms of the faulty and the healthy datasets, features can be extracted and classified for future use via the use of a Support Vector Machine (SVM). Therefore, boundary conditions can be formed for the under-examination parameters. As soon as the above-mentioned model is trained, then the model is constantly running and gets updated at a regular basis, given that there are new data posted on the Cloud database. In an attempt to generate a fault dataset, modification of the Simulink model is required. The modification involved the creation of additional subsystems which are used for the simulation of faults. Ultimately, the goal of this experiment series is to generate fault datasets, i.e. datasets containing measurements of the physical model operating under malfunction. For the generation of the fault datasets, a pressure drop in the refrigerant network/ piping was simulated and increased humidity within the cooling chamber. In order to process the data derived from the simulation runs, the outputs were transformed via Fourier Transformation, in order to represent the events in the frequency domain. Then, with the use of spectrograms, useful features were extracted and based on these features, with the use of a Support Vector Machine, the faults could be classified.

Augmented Reality Module

An AR module is provisioned in order to facilitate the monitoring process of the industrial equipment. This module can be realized as a multi-platform application, from which the customers can either remotely or on-site visualize crucial information about their equipment and interact with it, easily and intuitively with the use of this cutting-edge digital technology. Concretely, the current implementation of the framework supports handheld Android-based platforms, e.g. mobile phones and tablets, as well as Head Mounted Displays (HMD), such as the Microsoft HoloLens.

User tracking and pose estimation for the Android-based devices is based on the recognition of a feature-rich image target, as in the one presented in **Figure 5**. Further to that, in **Figure 5A** the physical form of the image target, whereas in **Figure 5B**, the features recognized by the device are overlaid on the image target.

As soon as the image target is recognized by the device, through the integrated camera, then the transformation matrix, denoted as T, between the camera and the marker is **Eq. 1**:

$$x_c = T * X \tag{1}$$

Where:

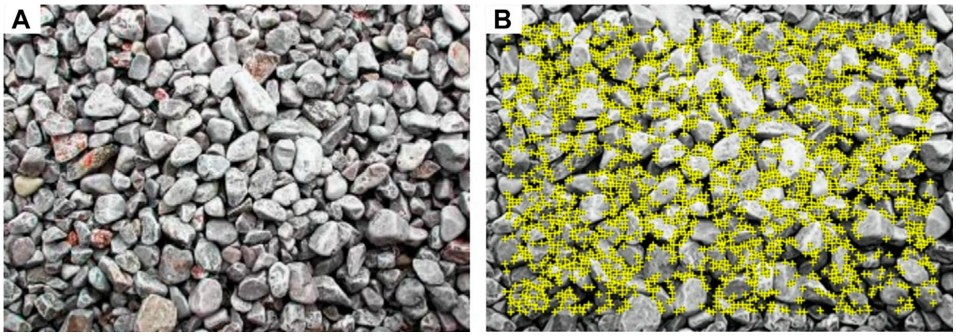


FIGURE 5 | Example of Image Target used in Android-based AR applications.

x_c is the projection of a point in ideal image coordinates.

T is the pose matrix.

X expresses the points in world coordinates.

Therefore for the calculation of x_c a 3×3 rotation matrix is utilized, denoted by R, as per the **Eq. 2**.

$$x_c = [R|t] * X \Rightarrow \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} r_{1,1} & r_{1,2} & r_{1,3} & t_x \\ r_{2,1} & r_{2,2} & r_{2,3} & t_y \\ r_{3,1} & r_{3,2} & r_{3,3} & t_z \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{2}$$

Afterwards, in order to translate the result in pixel coordinates, denoted by x_{pix}, i.e. as a 2D representation, calibration matrix is used, denoted by C. Consequently, x_{pix} based on **Eq. 3** becomes:

$$x_{pix} = K * x_c \Rightarrow \begin{pmatrix} x_{pix} \\ y_{pix} \\ 1 \end{pmatrix} = \begin{bmatrix} f & 0 & p_x & 0 \\ 0 & f & p_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} \tag{3}$$

Based on the pose estimation steps described in the previous paragraph, for the registration of the AR content using Android-based platforms, a fiducial image target is required. As soon as it is recognized by the device camera, then by calculating the user's position and pose, the augmentations are overlaid on the physical environment. What is more, in order to enhance the user experience, the application supports the functionality called "Extended Tracking", based on which, the handheld device can continue overlaying the augmentations in the physical environment in the event of the camera loosing direct contact with the image target.

However, for the implementation of the AR module in the Microsoft HoloLens HMD, the user pose estimation is approached in a different way, as the HMD is integrated with four (4) greyscale tracking cameras. As a result the Depth, denoted by D, is calculated with the use of **Eq. 4**.

$$D = \frac{R}{\sqrt{U^2 + V^2 + 1}} \tag{4}$$

Where:

D Is the Depth

R is the Range, which is measured from the integrated HoloLens ToF (Time-of-Flight) camera.

U and V are the distance values of a certain point.

Therefore the depth value into real, 3D world coordinates can be derived from **Eq. 5**.

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = D \begin{pmatrix} U \\ V \\ 1 \end{pmatrix} \tag{5}$$

Consequently, for the user pose estimation in the case of Microsoft HoloLens, the developed application initially prompts the user to select/setup an initial point of reference. This reference point is then automatically translated into a 3D world anchor based on which the AR visualizations are positioned and rendered in the physical environment.

One of the main aspects of the AR module/application, is the condition monitoring of the assets. This can be done in two ways. The first solution is for remote monitoring, where the responsible engineer uses either device to visualize a scaled 3D model of the refrigerator, upon which important information are displayed. The second solution is for monitoring the condition of the asset while inspecting it physically. In the case of the refrigerators, which is presented in the following paragraphs, it is impossible for the responsible engineer to monitor the current status/health of the refrigerator group without having to physically inspect the compressor unit, which is located away from the cooling chamber. However, with the proposed framework, it is possible to recognize the refrigerator, by utilizing the image targets, discussed previously and retrieve data for the corresponding refrigerator group from the Cloud database. As a result, the equipment/asset inspection can be performed in near-real-time.

Another aspect of the AR application is the provision of a communication tool, which enables the communication between the OEM and customer, in order to inspect the equipment in real time, and in addition to that to create basic AR instructions also in real time, by performing common "drag and drop" operation in the field of view of the user. Further to that, this functionality enables both the OEM and the client to communicate via a video call session, where the OEM can visualize the field of view of the user and with the use of basic 3D tool representation, the client can perform maintenance tasks in real time. The above-mentioned functionalities are based on the adaptation of the

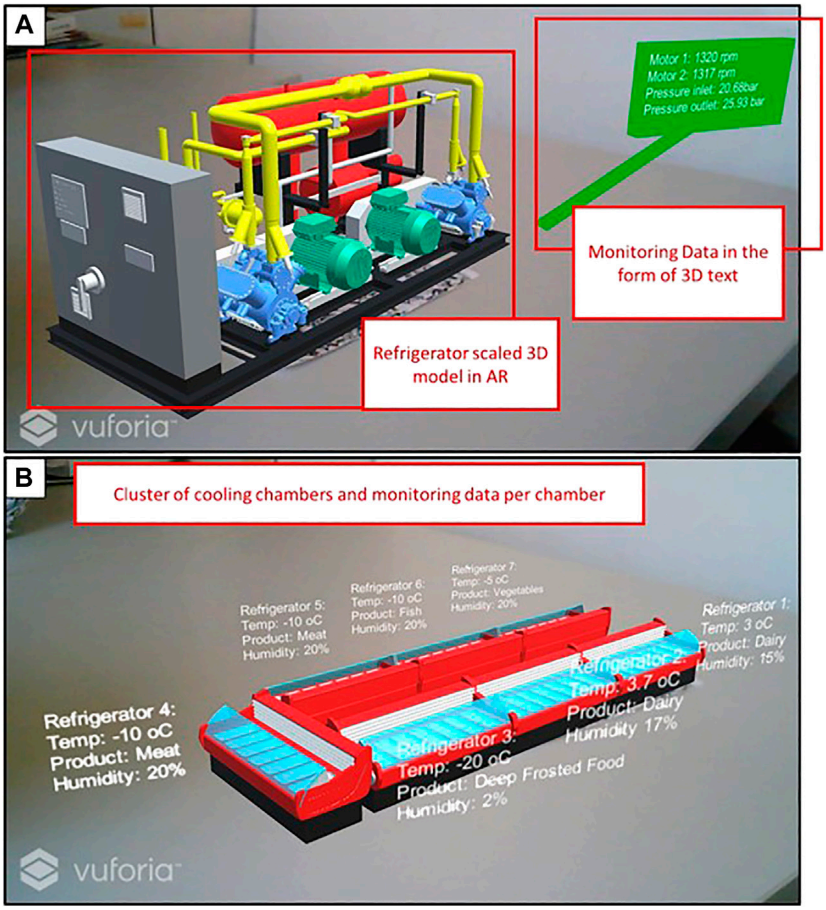


FIGURE 6 | (A) AR visualization of compressor and its working parameters; **(B)** Cooling chamber cluster and the current working conditions.

methodology presented in the research work of Mourtzis et al. (2020b) **Figure 6**.

SYSTEM IMPLEMENTATION

The proposed architecture can be realized as a multi-sided application. The first aspect of the application is a desktop-based application, which communicates with the server in order to retrieve the data from the server and process them through the predictive algorithm. The predictive algorithm is responsible for the identification of patterns within the processed data. Each of these patterns represents a classification of the possible situations of the under-examination machine, or cluster of machines. A predictive algorithm will have to analyze the data gathered from the sensors so that a prediction of unforeseeable machine malfunctions can be identified. However, since the data are available on the server, it is of great importance to create an application for monitoring the current situation of the machines.

From a software point of view, for the setup of the DAQ device, the Arduino IDE (Arduino, 2020) was used. Moreover, for the setup of the WSN the X-CTU (X-CTU, 2020) application from Digi has been utilized. For the development of the Graphical User Interfaces (GUI), a Universal Windows Platform (UWP) (Microsoft, 2018) application has been developed. The benefits of using UWP is the multi-platform implementation, the ease of configurability, ease of implementing security protocols, serviceability of the framework and updates' distribution. In the following paragraphs the functionalities and the GUIs designed and developed so far will be discussed in detail. In order to do so, the Unity 3D game engine is utilized (Unity, 2020). As regards the code scripts, the Microsoft Visual Studio IDE is used (Microsoft, 2020). More specifically, for the development of the main functionalities of the application, the code scripts are written in C# programming language. Since the application supports two user groups, one for the OEM supplier/service provider and one for the customers, a common login/register system is implemented. Upon installation of the application on the end-user's desired platform, the application prompts the user to register an account which automatically saved in the Cloud database. Therefore each time the user is connected, after getting authorized, their user group is automatically retrieved by the Cloud database and the suitable GUIs are loaded. It is stressed out that although the development of a UWP application enables multi-platform support, the AR functionalities are only available for handheld devices, such as Android-based mobile phones and

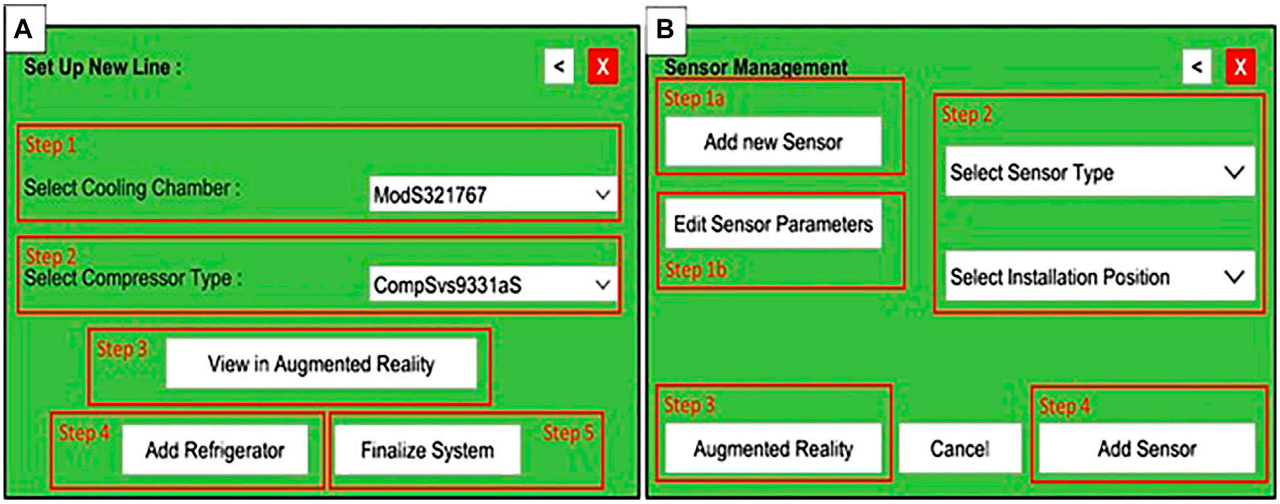


FIGURE 7 | (A) Settings for a new refrigerator line; **(B)** Settings for new/old sensors.

tablets, and Head Mounted Displays (HMD) such as Microsoft HoloLens.

As long as the login is successful, if the user is listed as a client, then the available options are to create a new cluster of refrigerators or process/monitor an existing cluster. The "Set Up New Line" functionality is targeted for new customers, or customers that acquired new equipment, i.e. new refrigerator group(s) or new DAQ device(s), as presented in **Figure 7A**.

The core functionality of the developed framework lies within the monitoring functionalities. In the corresponding GUIs, the customer can visualize in 3D a scaled version of their refrigerator group and upon request to visualize the available information, which are automatically fetched by the Cloud database. Then, in continuation, if requested, the data can be transformed into statistical figures, so that the client can visualize the current status of their equipment. All of the above-mentioned data can also be visualized in the form of augmentations in case the customer is close to the refrigerator. In order to further notify the customer about an upcoming maintenance action or if any piece of equipment requires special attention, certain alerts have been implemented as presented in **Figures 6** and **7**. Regarding the communication interfaces between the DAQ device, the end-user application and the Cloud Platform, RESTful API services have been developed. As regards the communication interface between the DAQ device and the Cloud Database, the DAQ device as a client can send POST, and PUT requests to the Cloud Platform, so as to enable the data to be posted on the database if they are not existent, or updated as needed. On the contrary, the majority of the services implemented on the end-user application send only GET requests, in order to fetch data to the end-user's device, with the exception of user registration, where a POST request is sent in order to record the user's data on the database. However, due to the large volumes of sensitive data that are circulated within the proposed framework, a set of security measures are taken. Initially, all the HTTP requests are of type HTTPS (Hypertext Transfer Protocol Secure). Secondly, DELETE requests are not allowed for anyone trying to connect to the Cloud Platform. Therefore, in order to delete any records from the database, this process has to be undertaken manually by the authorized system administrator. In an attempt to make the proposed framework more general, a custom editor has been developed for supporting the functionalities of the framework itself. More specifically, the development team assisted by the editor can create virtually any configuration of systems and functionalities, so that the framework can be adapted to the actual needs of the corresponding company.

From a hardware point of view, a desktop PC has been utilized for the development of the application as well as the Cloud Database and its services. For the implementation of the developed AR-based application, an Android-based tablet and a Microsoft HoloLens HMD are used. As regards, the DAQ device, an Arduino Mega 2560 microcontroller is paired with the custom board presented in the previous paragraphs. In addition to that for the XBee RF modules, the Arduino XBee shield and the Adafruit explorer shield were used.

CASE STUDY

In manufacturing systems the profit is derived by the subtraction of operating costs from total income. Therefore, the profit becomes a problem with two possible solutions, either the minimization of operating costs, or the maximization of income. As regards, operating costs, industrial equipment maintenance costs are also included. Production equipment affects operating costs with machine deterioration and failures as well.

The applicability of the developed framework has been tested and validated in a real-life industrial scenario,derived from an OEM supplier of professional refrigerator systems. The OEM is looking forward to transforming their business model based on

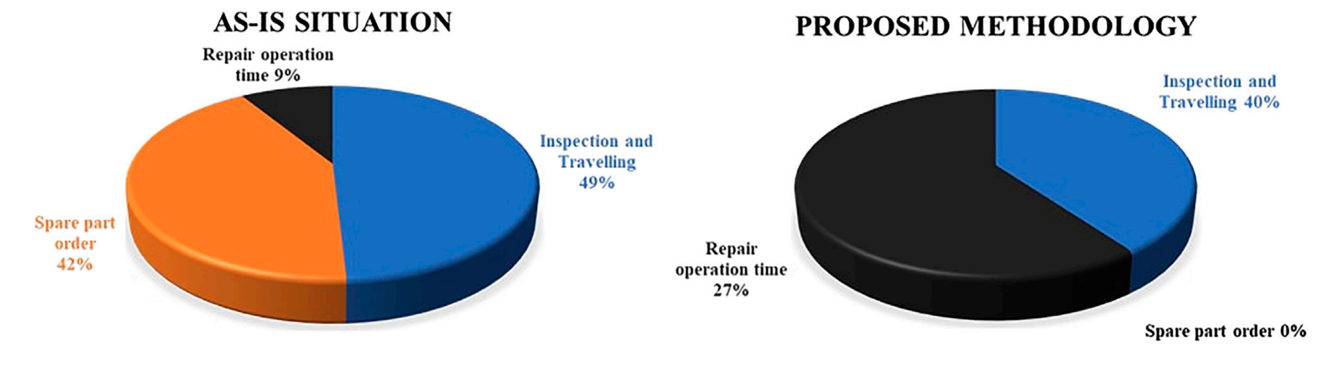


FIGURE 8 | Time distribution per task, as-is situation vs proposed

the PSS paradigm, by providing the proposed framework as a service, in an attempt to improve their after-sales policy. What is more, it is estimated that based on the monitoring and analysis from the AI algorithms, the engineering department of the OEM will be able to gather insightful feedback, aiming at the improvement of the design process and the quality of their products. The main benefit of the proposed methodology, is that it can provide time estimations about future equipment malfunctions, which by extension can enable both the OEM and the client to act proactively and in time, in order to further minimize the equipment downtime. Further to that, with the provision of the AR application knowing beforehand the upcoming equipment failures, can facilitate maintenance engineers to prepare the AR content timely and communicate it to the client. Therefore, the need for an external maintenance technician is further minimized, thus reducing both the overall time and cost of maintenance.

In order to test and validate the proposed framework, the DAQ device was installed, on an experimental refrigerator group located at the premises of the OEM, used for test purposes. It is stressed out that the OEM has already integrated sensing systems on the majority of their products for monitoring purposes. However, the existing solution is wired and requires a computer and an engineer close to the refrigerator compartment, in order to monitor their status. Therefore, the first step was the installation of the DAQ device, presented in the previous paragraphs as well as the setup of the required WSN network. The WSN follows the star topology, meaning that the one XBee is connected to a PC and acts as the WSN coordinator. Then, each DAQ connects to the coordinator and transmits the data at the defined rate. Afterwards, in order to handle the data arrived at the WSN coordinator, the corresponding COM port is listened by the PC via a Python script and the data are uploaded to the database and saved within the corresponding CSV file. Then the data saved in the Cloud database are automatically input to the Digital Twin of the refrigerator in order to simulate its condition and calculate the RUL for the critical components. Based on this setup the monitoring and simulation runs were executed at the premises of the case study provider. For the purposes of the experiments, a scenario of compressor malfunction has been examined. For the maintenance of the refrigerator in such case, the inspection of the equipment from an OEM technician would require travelling and inspection which according to the OEM it would account for approximately 28 h, the order and acquisition of the replacement compressor would require approximately 24 h and the installation on the refrigerator group would require approximately 5 h. However, with the adoption of the proposed framework, and based on the calculation of the compressor RUL, the client is capable to order the failing part timely, thus minimizing the waiting time. Further to that, with the utilization of the AR application the inspection of the equipment can be facilitated, thus eliminating the need for an OEM maintenance technician to visit the client. The time for inspection was calculated to be approximately 5 h. In the following figure, the time estimations for the current situation as well as the corresponding times with the adoption of the proposed methodology are presented **Figure 8**.

CONCLUDING REMARKS AND OUTLOOK

The scope of this research work was to present the latest trends regarding the fields of predictive maintenance and XR and furthermore to propose a novel framework that will facilitate engineers to constantly monitor the status of the manufacturing equipment and in advance to predict the forthcoming maintenance activities. By extension, the prediction of malfunctions will enable companies to schedule their production more efficiently, whilst it makes them more adaptive to any disturbances caused within the company limits. From the practical implementation of the developed framework in the industrial partner, it became evident that the refrigerator downtimes can be reduced by approximately 20%, since both the clients and the OEM were capable to monitor the status of their equipment and by extension, with the use of the AI algorithm, the RUL prediction for crucial components of the refrigerator system, the client got a trustworthy estimation of when their equipment should be maintained. In addition to that, since the customer, could get an estimation of the upcoming failure, they are able to schedule a maintenance session with the OEM much faster and fitted to both ends' schedule without creating great disturbances. An equally important finding is that the maintenance costs can be reduced by approximately 10% since the OEM can order and acquire the needed components beforehand, thus eliminating overnight delivery costs. Along with that, by predicting and scheduling timely the maintenance session, the equipment is not left to run until failure, which could affect the operation of other components, thus leading to increased maintenance cost, due to additional technician labor and extra replacement parts.

Although the development and the implementation of the proposed framework have yielded promising results, there are several implications that must be addressed before such solutions reach an acceptable maturity level and by extension, become commercially available. The most important implication faced, is that the calculation of the RUL cannot be performed in real-time thus inducing a certain amount of latency in the AR visualizations. The amount of latency is affected by two major issues, first the network speed and second the computational power of the system handling the Digital Twin. Another implication is the authoring of the AR visualizations in the field of view of the end-users. Currently, this is a manual task, which in the future must be addressed, so that AR applications can become useful tools in the modern manufacturing environment rather than increasing the complexity of the systems, by increasing the time and effort to prepare the so-called "AR scenes".

In the future, the Digital Twin will also be improved. It is estimated that following the implementation of the proposed framework in similar equipment, i.e. a fleet of assets, will enable the creation of data ensembles. The idea behind this is to utilize similar datasets in order to improve the predicting accuracy of the Digital Twin.

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

AUTHOR CONTRIBUTIONS

MD conceived the idea and supervised the writing and the experimentation of the research work, AJ is responsible for the conceptualization of the project and the writing of the paper, PN conducted the research for similar works and contributed in the results as well as in the writing of the paper.

FUNDING

This work has been partially supported by the H2020 EC funded project "SPARE – Sensor-based Product monitoring system to support Augmented Reality remote maintenancE" (GA No: 80765 ΔΕΡ5-0020271).

REFERENCES

Ahmed, M., and Khan, A.-S.P. (2019). *Data analytics: concepts, techniques, and applications*. Boca Raton, FL: CRC Press Taylor & Francis Group.

Arduino. (2020). Available at: https://www.arduino.cc/ (Accessed October 2,2020)

Banerjee, I., Ling, Y., Chen, C. M., Hasan, A. S., Langlotz, P. C., Moradzadeh, N., et al. (2019). Comparative effectiveness of convolutional neural network (CNN) and recurrent neural network (RNN) architectures for radiology text report classification. *Artif. Intell. Med.*, 97:79–88. doi:10.1016/j.artmed.2018.11.004

Baru, A. (2018). Three ways to estimate remaining useful life for predictive maintenance. Technical Articles and Newsletters, MathWorks. Available at: https://www.mathworks.com/company/newsletters/articles/three-ways-to-estimate-remaining-useful-life-for-predictive-maintenance.html (Accessed June 20, 2020)

Chen, J., Jing, H., Chang, Y., and Liu, Q. (2019). Gated recurrent unit based recurrent neural network for remaining useful life prediction of nonlinear deterioration process. *Reliab. Eng. Syst. Saf.* 185:372–382. doi:10.1016/j.ress.2019.01.006

Chryssolouris, G. (2006). *Manufacturing systems: theory and practice*, 2nd ed. New York, NY: Springer-Verlag

Deloitte, (2017a). Predictive maintenance and the smart factory, Available at: https://www2.deloitte.com/content/dam/Deloitte/us/Documents/process-and-operations/us-cons-predictive-maintenance.pdf (Accessed June 20, 2020)

Deloitte, (2017b). Predictive Maintenance. Taking pro-active measures based on advanced data analytics to predict and avoid machine failure, Analytics Institute. Available at: https://www2.deloitte.com/content/dam/Deloitte/de/Documents/deloitte-analytics/Deloitte_Predictive-Maintenance_PositionPaper.pdf (Accessed June 20, 2020)

Egger, J., and Masood, T. (2020). Augmented reality in support of intelligent manufacturing – a systematic literature review. *Comput. Ind. Eng.* 140, 106195. doi:10.1016/j.cie.2019.106195

Erfani, S. M., Rajasegarar, S., Karunasekera, S., and Leckie, C. (2016). High-dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning. *Pattern Recogn.* 58, 121–134. doi:10.1016/j.patcog.2016.03.028

Fantini, P., Pinzone, M., and Taisch, M. (2020). Placing the operator at the centre of Industry 4.0 design: modelling and assessing human activities within cyber-physical systems. *Comput. Ind. Eng.* 139, 105058. doi:10.1016/j.cie.2018.01.025

Fast-Berglund, Å., Gong, L., and Li, D. (2018). Testing and validating Extended Reality (xR) technologies in manufacturing. *Procedia Manufacturing*. 25, 31–38. doi:10.1016/j.promfg.2018.06.054

Hupfer, S. (2019). Capitalizing on the promise of artificial intelligence. Perspectives on AI adoption from around the world, Deloitte Insights. Available at: https://www2.deloitte.com/us/en/insights/focus/cognitive-technologies/global-perspectives-ai-adoption.html (Accessed June 20, 2020)

Kong, Z., Cui, Y., Xia, Z., and He, L. (2019). Convolution and long short-term memory hybrid deep neural networks for remaining useful life prognostics. *Appl. Sci.* 9(19), 4156–4174. doi:10.3390/app9194156

Kozjek, D., Malus, A., and Vrabi", R. (2020). Multi-objective adjustment of remaining useful life predictions based on reinforcement learning. *Procedia CIRP*. 93, 425–430. doi:10.1016/j.procir.2020.03.051

Lei, Y., Li, N., Guo, L., Li, N., Yan, T., and Lin, J. (2018). Machinery health prognostics: a systematic review from data acquisition to RUL prediction. *Mech. Syst. Signal Process.* 104, 799–834. doi:10.1016/j.ymssp.2017.11.016

Li, X., Zhang, W., and Ding, Q. (2019). Deep learning-based remaining useful life estimation of bearings using multi-scale feature extraction. *Reliab. Eng. Syst. Saf.* 182, 208–218. doi:10.1016/j.ress.2018.11.011

Li, X., Ding, Q., and Sun, J. Q. (2018). Remaining useful life estimation in prognostics using deep convolution neural networks. *Reliab. Eng. Syst. Saf.* 172, 1–11. doi:10.1016/j.ress.2017.11.021

Liu, Y., Hu, X., and Zhang, W. (2019). Remaining useful life prediction based on health index similarity, *Reliab. Eng. Syst. Saf.* 185, 502–510. doi:10.1016/j.ress.2019.02.002

Loutas, T. H., Roulias, D., and Georgoulas, G. (2013). Remaining useful life estimation in rolling bearings utilizing data-driven probabilistic E-support vectors regression. *IEEE Trans. Reliab.* 62(4), 821–832. doi:10.1109/TR.2013.2285318

Masoni, R., Ferrise, F., Bordegoni, M., Gattullo, M., Uva, A. E., Fiorentino, M., et al. (2017). Supporting remote maintenance in industry 4.0 through augmented reality. *Procedia Manufacturing*. 11, 1296–1302. doi:10.1016/j.promfg.2017.07.257

Microsoft. (2020). Visual Studio. Best-in-class tools for any developer. Available at: https://visualstudio.microsoft.com/ (Accessed June 20, 2020)

Microsoft. (2018). What's a universal Windows platform (UWP) app? Available at: https://docs.microsoft.com/en-us/windows/uwp/get-started/universal-application-platform-guide (Accessed June 20, 2020)

Mitchell, R. J. (1995). "How computers process data." In: *Microprocessor systems*. London: Palgrave.

Mourtzis, D. (2020). Simulation in the design and operation of manufacturing systems: state of the art and new trends. *Int. J. Prod. Res.* 58(7), 1927–1949. doi:10.1080/00207543.2019.1636321

Mourtzis, D., Angelopoulos, J., and Panopoulos, N. (2020a). A framework for automatic generation of augmented reality maintenance & Repair instructions based on convolutional neural networks, 53rd CIRP conference on manufacturing systems (CMS 2020). *Procedia CIRP*. 93, 977–982. doi:10.1016/j.procir.2020.04.130

Mourtzis, D., Siatras, V., Angelopoulos, J., and Panopoulos, N. (2020c). An augmented reality collaborative product design cloud-based platform in the context of learning factory. *Procedia Manufacturing*. 45, 546–551. doi:10.1016/j.promfg.2020.04.076

Mourtzis, D., Siatras, V., and Angelopoulos, J. (2020b). Real-time remote maintenance support based on augmented reality (AR). *Appl. Sci.* 10(5). doi:10.3390/app10051855

Mourtzis, D., Vlachou, A., and Zogopoulos, V. (2017). Cloud-based augmented reality remote maintenance through shop-floor monitoring: a product-service system approach. *J. Manuf. Sci. Eng.* 139(6), 061011. doi:10.1115/1.4035721

Mourtzis, D., Vlachou, E., Milas, N., and Xanthopoulos, N. (2015). A cloud-based approach for maintenance of machine tools and equipment based on shop-floor monitoring. *Procedia CIRP* 41, 655–660. doi:10.1016/j.procir.2015.12.069

Mourtzis, D., Zogopoulos, V., Katagis, I., and Lagios, P. (2018). Augmented reality based visualization of CAM instructions towards industry 4.0 paradigm: a CNC bending machine case study. *Procedia CIRP* 70, 368–373. doi:10.1016/j.procir.2018.02.045

Nicholson, A., Webber, S., Dyer, S., Patel, T., and Janicke, H. (2012). SCADA security in the light of cyber-warfare. *Comput. Secur.* 31(4), 418–436. doi:10.1016/j.cose.2012.02.009

Palmarini, R., Erkoyuncu, A. J., Roy, R., and Torabmostaedi, H. (2018). A systematic review of augmented reality applications in maintenance. *Robot. Comput. Integrated Manuf.* 49, 215–228. doi:10.1016/j.rcim.2017.06.002

Pang, Y., He, Q., Jiang, G., and Xie, P. (2020). Spatio-temporal fusion neural network for multi-class fault diagnosis of wind turbines based on SCADA data. *Renew. Energy*. 161, 510–524. doi:10.1016/j.renene.2020.06.154

Rezaeianjouybari, B., and Shang, Y. (2020). Deep learning for prognostics and health management: state of the art, challenges, and opportunities. *Measurement*. 163, 107929. doi:10.1016/j.measurement.2020.107929

Ruiming, F., Minling, W., Xinhua, G., Rongyan, S., and Pengfei, S. (2020). Identifying early defects of wind turbine based on SCADA data and dynamical network marker. *Renew. Energy* 154, 625–635. doi:10.1016/j.renene.2020.03.036

Sun, H., Liu, Y., Pan, J., Zhang, J., and Ji, W. (2020). Enhancing cutting tool sustainability based on remaining useful life prediction, *J. Clean. Prod.* 244, 118794. doi:10.1016/j.jclepro.2019.118794

Tao, F., Qi, Q., Liu, A., and Kusiak, A. (2018). Data-driven smart manufacturing. *J. Manuf. Syst.* 48, 157–169. doi:10.1016/j.jms y.2018.01.006

Unity. (2020). Available at: https://unity.com/ (Accessed June 20, 2020)

Vogl, G. W., Weiss, B. A., and Helu, M. (2019). A review of diagnostic and prognostic capabilities and best practices for manufacturing. *J. Intell. Manuf.* 30(1), 79–95. doi:10.1007/s10845-016-1228-8

Vorraber, W., Gasser, J., Webb, H., Neubacher, D., and Url, P. (2020). Assessing augmented reality in production: remote-assisted maintenance with HoloLens. *Procedia CIRP*. 88, 139–144. doi:10.1016/j.procir.2020.05.025

Wang, H., Peng, M. J., Miao, Z., Liu, Y. K., Ayodeji, A., and Hao, C. (2020). Remaining useful life prediction techniques for electric valves based on convolution auto encoder and long short term memory. *ISA (Instrum. Soc. Am.) Trans.* doi:10.1016/j.isatra.2020.08.031

Wen, P., Zhao, S., Chen, S., and Li, Y. (2021). A generalized remaining useful life prediction method for complex systems based on composite health indicator. *Reliability Engineering & System Safety*. 205, 107241. doi:10.1016/j.ress.2020.107241

Wolfartsberger, J., Zenisek, J., and Wild, N. (2020). Data-driven maintenance: combining predictive maintenance and mixed reality-supported remote assistance. *Procedia Manufacturing*. 45, 307–312, doi:10.1016/j.promfg.2020.04.022

X-CTU, (2020). Available at: https://www.digi.com/products/embedded-systems/digi-xbee/digi-xbee-tools/xctu (Accessed October 2, 2020)

Yang, F., Habibullah, S. M., and Shen, Y. (2021). Remaining useful life prediction of induction motors using nonlinear degradation of health index. *Mech. Syst. Signal Process*. 148, 107183. doi:10.1016/j.ymssp.2020.107183

Yang, H., Zhao, F., Jiang, G., Sun, Z., and Mei, X. (2019). A novel deep learning approach for machinery prognostics based on time Windows. *Appl. Sci.* 9, 4813. doi:10.3390/app 9224813

Zhang, S., and Lang, Z. Q. (2020). SCADA-data-based wind turbine fault detection: a dynamic model sensor method. *Contr. Eng. Pract.* 102, 104546. doi:10.1016/j.conengprac.2020.104546

Zhao, Z., Liang, B., Wang, X., and Lu, W. (2017). Remaining useful life prediction of aircraft engine based on degradation pattern learning, *Reliab. Eng. Syst. Saf.* 164, 74–83. doi:10.1016/j.ress.2017.02.007

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

frontiers in Artificial Intelligence

ORIGINAL RESEARCH
published: 15 February 2021
doi: 10.3389/frai.2020.570562

RECLAIM: Toward a New Era of Refurbishment and Remanufacturing of Industrial Equipment

Angeliki Zacharaki[1]*, *Thanasis Vafeiadis*[1], *Nikolaos Kolokas*[1], *Aikaterini Vaxevani*[1], *Yuchun Xu*[2], *Michael Peschl*[3], *Dimosthenis Ioannidis*[1] and *Dimitrios Tzovaras*[1]

[1]Information Technologies Institute, Center for Research and Technology Hellas, Thessaloniki, Greece, [2]College of Engineering and Physical Sciences, Aston University, Birmingham, United Kingdom, [3]Harms and Wende GmbH and Co., Hamburg, Germany

OPEN ACCESS

Edited by:
Dimitris Kiritsis, École Polytechnique Fédérale de Lausanne, Switzerland

Reviewed by:
Pankush Kalgotra, Auburn University, United States
Ajith Kumar Parlikad, University of Cambridge, United Kingdom

***Correspondence:**
Angeliki Zacharaki
angezach@iti.gr

Specialty section:
This article was submitted to AI in Business, a section of the journal Frontiers in Artificial Intelligence

Received: *08 June 2020*
Accepted: *16 November 2020*
Published: *15 February 2021*

Citation:
Zacharaki A, Vafeiadis T, Kolokas N, Vaxevani A, Xu Y, Peschl M, Ioannidis D and Tzovaras D (2021) RECLAIM: Toward a New Era of Refurbishment and Remanufacturing of Industrial Equipment. Front. Artif. Intell. 3:570562. doi: 10.3389/frai.2020.570562

Refurbishment and remanufacturing are the industrial processes whereby used products or parts that constitute the product are restored. Remanufacturing is the process of restoring the functionality of the product or a part of it to "as-new" quality, whereas refurbishment is the process of restoring the product itself or part of it to "like-new" quality, without being as thorough as remanufacturing. Within this context, the EU-funded project RECLAIM presents a new idea on refurbishment and remanufacturing based on big data analytics, machine learning, predictive analytics, and optimization models using deep learning techniques and digital twin models with the aim of enabling the stakeholders to make informed decisions about whether to remanufacture, upgrade, or repair heavy machinery that is toward its end-of-life. The RECLAIM project additionally provides novel strategies and technologies that enable the reuse of industrial equipment in old, renewed, and new factories, with the goal of saving valuable resources by recycling equipment and using them in a different application, instead of discarding them after use. For instance, RECLAIM provides a simulation engine using digital twin in order to predict maintenance needs and potential faults of large industrial equipment. This simulation engine keeps the virtual twins available to store all available information during the lifetime of a machine, such as maintenance operations, and this information can be used to perform an economic estimation of the machine's refurbishment costs. The RECLAIM project envisages developing new technologies and strategies aligned with the circular economy and in support of a new model for the management of large industrial equipment that approaches the end of its design life. This model aims to reduce substantially the opportunity cost of retaining strategies (both moneywise and resourcewise) by allowing relatively old equipment that faces the prospect of decommissioning to reclaim its functionalities and role in the overall production system.

Keywords: refurbishment and remanufacturing, decision support framework, *in situ* repair, big data analytics, predictive analytics, industry, machine learning

1. INTRODUCTION

The industrial sector in Europe is very important as a "driver of sustainable growth and employment" (EP (2019)). High industrial productivity and efficiency are closely linked to well-functioning and well-maintained equipment, thus highlighting the critical role of machinery. However, not only in Europe is industry so vitally important for the economy. Currently, there is an estimated 40 billion dollars' worth of outdated machinery in use at US factories (M.NET (2016)), leading to an estimated loss of 50 billion dollars each year due to unplanned downtime resulting from machine failures (Studios (2018)). This is to some extent expected since many machines currently in use in production lines were installed well over 30 years ago and have exceeded their projected lifetime. In order to remain competitive, manufacturing companies should constantly improve the productivity and reliability of their production processes and equipment. In this perspective, maintenance activities have become even more crucial for business success. It is worth noticing that, nowadays, poor maintenance strategies reduce the industry's overall capacity between 5 and 20 percent (Wollenhaupt (2017)). This underlines the urgent need for improving the maintenance process, emphasizing the methods of refurbishment and remanufacturing. Refurbishment and remanufacturing are activities of the circular economy model, the purpose of which is to keep the high value of products and materials, as opposed to the currently employed economic model, thus targeting the extension of equipment and materials' life and reducing the unnecessary and wasteful use of resources. These two activities, along with health status monitoring, are the key elements for lifetime extension and reuse of large industrial equipment.

The EU Factory of the Future project, RECLAIM (*RE-manufaCturing and Refurbishment LArge Industrial equipment*), focuses on establishing and demonstrating technologies and techniques in order to support a new approach for refurbishment and remanufacturing of large industrial equipment in factories, setting forth the way to a circular economy. The project's main aim is to improve the maintenance process, emphasizing the methods of refurbishment and remanufacturing. Its ultimate objective is to preserve valuable resources by reusing equipment instead of discarding it. In this context, the project will develop new models and strategies for repairing and upgrading equipment and redesigning factory layouts to benefit the manufacturing sector from an economic perspective. These strategies are as follows: improving the machinery operation and avoiding unplanned downtime due to machine failure; estimating life cycle costs and contributing to the reuse of old machinery assets in renewed and new factories; providing maintenance able to identify equipment failures before they occur, in order to minimize the additional costs and downtime associated with the disassembly and transportation of the machinery and maximize the performance of the machinery during its lifetime.

The main scope of this work is to present the new paradigm for refurbishment and remanufacturing of large industrial equipment in factories, paving the way to a circular economy. In particular, firstly, we conducted a bibliographic review of the technologies, strategies, and tools that have been used to date to achieve the refurbishment and remanufacturing of the large industrial equipment in order to extend its lifetime. Then, the usefulness of the integrated technological solution RECLAIM was analyzed to prove that having RECLAIM technology available drastically increased efficiency, enhanced lifetime extension, and achieved high economic benefit and a significant step toward 100% reuse will be made. The added value of this article is to contribute to a better understanding of how the integration of RECLAIM technological solutions into industrial environments can lead to industries having an extra economic income from the extended lifetime of manufacturing systems and their components, which can be achieved by adopting refurbishment and remanufacturing solutions.

The rest of this article is organized as follows. In **Section 2**, the related work regarding key aspects that the RECLAIM project will face along with the proposed solutions dealing with these problems is analyzed, whereas the conceptual architecture is presented in **Section 3**. In **Section 4**, the main components of the architecture are described in full, and, finally, in **Section 5**, our conclusions are drawn.

2. RELATED WORK AND PROPOSED SOLUTIONS

In this section, in the beginning, the related works on key aspects of refurbishment and remanufacturing in the manufacturing environment are presented along with the ambition and solutions proposed by the project. The limitations of the RECLAIM platform are given at the end of the current section.

2.1. RECLAIM Solutions

The RECLAIM integrated architecture encompasses several modules and components, which are described in the following sections. These components are as follows: (a) decision support framework (DSF), (b) refurbishment and remanufacturing techniques for industrial equipment, (c) smart sensors' network for industrial environments, (d) prognosis and health management, (e) cost analysis and cost modeling, (f) optimization planning on refurbishment and remanufacturing, (g) digital twin simulation engine, (h) cybersecurity for IoT devices, and (i) augmented reality (AR).

2.1.1. Decision Support Framework of Used Industrial Equipment for Sustainable Manufacturing

Several methodologies and approaches for decision-making have been developed, addressing either single or multiple decisions, in order to enable the users and the process experts to assess the reusability or remanufacturability of production machines when they are facing the end of their production life. Ziout et al. (2014) provided a decision-making methodology for considering the end-of-life (EoL) product as a recovery option for all interested parties involved in the process. Thus, the decision taken is more accurate and more informative on the selection of appropriate recovery options. Remery et al. (2012) have provided an EoL scenario assessment methodology to evaluate the various solutions for the EoL scenario of a product during the early design phase, based on fuzzy techniques. Dhouib (2014) proposed a multicriteria decision analysis based on an extended (fuzzy) version of Measuring Attractiveness by a Categorical Based Evaluation Technique (MACBETH) methodology to take into consideration the imprecise and linguistic assessments provided by a decision-maker. Ovchinnikov et al. (2014) presented an analytical model and a behavioral study to demonstrate that remanufacturing frequently aligns the economic and environmental objectives of firms by increasing profits and reducing their overall impact on the environment. Ondemir and Gupta (2014) developed a multiple objective advanced order remanufacturing and disassembly (ARTODTO) system as an order-driven component and product recovery (ODCPR) system. The main objective of the proposed system was to evaluate whether a product needs remanufacturing, disassembly, repair, or recycling and to accomplish a variety of conflicting financial, environmental, and quality-based goals. Ng and Song (2015) introduced a method for determining the optimal for EoL product recovery options, using multiple factors of product condition and recovery values aiming to achieve better environmentally friendly decision-making regarding the maintenance, repair, or remanufacturability planning of the product. Dehghanbaghi et al. (2016) proposed an integrated approach, based on a fuzzy rule-based system and a fuzzy Analytic Hierarchy Process (AHP), to provide an accurate and valid decision-making mechanism to rank the recovery/ disposal strategies. Kremer (2015) demonstrated a framework based on fuzzy logic and analysis that takes into account quantitative indicators (residual value and ecoindicators) and the social impact of each EoL option of the machinery.

Table 1 presents the key requirements for effective remanufacturing decision-making that are presented above compared to the aspects that RECLAIM DSF (*Decision Support Framework)* take into account.

The aforementioned studies introduced several decision-making methodologies and frameworks that utilize fuzzy logic and include background of the product, environmental criteria, and stakeholders with the intention of providing a reasonable decision for the recovery or rejection of the product. Nevertheless, few of them could help to achieve both machine- and component-level health-based recovery planning while taking into account at the same time both environmental and economic reverberations. Thus, RECLAIM aims to develop a flexible recovery DSF to assist the machinery operators and machinery manufacturers in making efficient remanufacturing, repair, or rejection decisions at different service life periods on electromechanical machines and robotic systems. The proposed solution will combine inputs from optimization techniques, machine learning, digital twins, fault diagnosis, and predictive maintenance simulations.

2.1.2. Uses of Refurbishment and Remanufacturing in Industrial Equipment

Refurbishment and remanufacturing can contribute significantly to well-being in Europe, as they are important lifetime extension strategies of resource-efficient manufacturing. In particular, the refurbishment process restores the system to meet its original specifications without replacing parts of the system (Varde et al. (2014)). Numerous methods have been used as refurbishment processes in a variety of industrial sectors (e.g., automotive, electrical, and electronics) in order to prolong the remaining useful life (RUL) of industrial infrastructure (Freiberger et al. (2011); Hatcher et al. (2013)). Atasu et al. (2008) have identified that the refurbishment process can be used as a productivity enhancement measure and as a marketing strategy. In this direction, Kerr and Ryan (2001) have reported that the refurbishment process reduces the total life cycle cost of the industrial infrastructure and is an ecofriendly method. However, many works have identified that the extension of the product life can also be achieved by using the remanufacturing process, which is also a value recovery option in order to extend industrial equipment's original lifespan (Chari et al. (2014)). In the work conducted by SteingrímssonPinar et al. (2011), the authors have introduced competitive business approaches regarding the remanufacturing market of production equipment. In a similar work, Cunha et al. (2011) have established a technology roadmapping methodology so as to portray the interconnections between market, equipment, and technology variables in the remanufacturing process. Moreover, a different make-to-order production strategy for automotive equipment was introduced by Schraven et al. (2012). In this work, the authors proposed a modular concept that enables consideration of recovered equipment components in engineering and design. Sharma et al. (2015) have explored the benefits of the remanufacturing process and showed that this process meets the requirements and needs of industries. In a more contemporary work, Darghouth et al. (2017) focused on the requirements of OEM perspectives for an effective remanufacturing process. Finally, the usefulness of the remanufacturing process was identified in many research works that were devoted either to machine tools (Ullah et al. (2016)) or to a special kind of production machines (Geng et al. (2016)).

With RECLAIM, a set of novel tools and methodologies will be developed for enhancing the refurbishment and remanufacturing processes for industrial electromechanical machines and robotic systems, differentiating the approach according to the level of action (whole machine, modules. and components). This approach is aiming not only to improve refurbishment and remanufacturing processes but also to participate in effective decision-making so as to achieve measurable performance improvements. The adaptive sensor network and digital retrofitting infrastructure will have the ability to be attached to machines, modules, and even components to be refurbished or remanufactured. This will allow legacy equipment to be a part of the IoT with advanced vertical and horizontal communication capabilities and also enable sophisticated data analysis, such as predictive maintenance. Those services will be supported by AR mechanisms (AR glasses) that will help technicians and manufacturers with a novel way to visualize and localize information on equipment refurbishment and remanufacturing operations directly situated on top of the physical equipment.

TABLE 1 | Key requirements for effective remanufacturing decision-making.

Author(s)	Key requirements for effective remanufacturing decision-making	Limitations
Ziout et al. (2014)	Recovery options for end-of-life products	Costs, environmental and health safety, durability, and viability of upgraded machine
Remery et al. (2012)	Fuzzy techniques during the early product design phase	Circular economy strategies
Dhouib (2014)	Extended (fuzzy) version of Measuring Attractiveness by a Categorical Based Evaluation Technique (MACBETH) methodology	Circular economy strategies, recovery options, durability, and viability of upgraded machine
Ovchinnikov et al. (2014)	Economic and environmental objectives	Circular economy strategies, recovery options, durability, and viability of upgraded machine
Ondemir and Gupta (2014)	Multiple objective advanced order remanufacturing and disassembly (ARTODTO) system as an order-driven component and product recovery (ODCPR) system	Durability
Ng and Song (2015)	Environmentally friendly decision-making	Circular economy strategies, recovery options, durability, and viability of upgraded machine
Dehghanbaghi et al. (2016)	Fuzzy rule-based system and a fuzzy AHP	Recovery options
Kremer (2015)	Quantitative indicators (residual value, ecoindicators, etc.) and the social impact	Circular economy strategies, recovery options, durability, and viability of upgraded machine

2.1.3. Smart Sensor Network for Industrial Environment

Applications in the manufacturing domain, regarding real-time monitoring, maintenance planning, fault detection, etc., based on sensor network and data have yet to be widely adopted. Mourtzis et al. (2014b) focused on the integration of the customer into product personalization and aimed to support the design of manufacturing networks on the move through the development of apps for Android devices. Mori and Fujishima (2013) introduced a remote monitoring and maintenance system for machine tool manufacturers that uses a mobile phone. Tapoglou et al. (2015) adopted a cloud-based approach for monitoring the use of manufacturing equipment through a network of sensors dispatching assignments to designated computer numerical control machines and generating the optimum machining code. Teti et al. (2010) demonstrated that the primary requirements for sensorial monitoring systems in production involving sensors of any type (mostly vibration, acoustic, and temperature) are robustness, reconfiguration capability, intelligence, reliability, and cost-efficiency. Si et al. (2011) pointed out the increasing interest in the use of multiple sensors for condition monitoring and illustrated various multisensor data fusion methods and recent developments in diagnostics and prognostics of mechanical systems implementing condition-based maintenance. Rehorn et al. (2005) presented sensor and signal processing techniques used for tool condition monitoring systems in industrial machining applications. Moreover, D. Mourtzis et al. (2014) have proposed a framework of machine monitoring techniques for almost real-time machine status recognition that facilitates a predictive maintenance engine to minimize machine tool failures.

In order to provide automated knowledge extraction from big data gathered by the industrial partners, RECLAIM will use smart real-time control and data analytics, monitoring forecasted production lines and allowing prescriptive and preventative actions. RECLAIM will enable "just-in-time manufacturing" to continuously adjust to the business environment. The analytic tools will allow to easily adapt a) the quantity and variability of heterogeneous information sources across the factory life cycle, b) the different types of targeted manufacturing sectors and their structure, and c) the differences in the enterprise hierarchy level.

2.1.4. Prognostics and Health Management Approach

To determine if a machine is worth refurbishing or remanufacturing, a Prognostics and Health Management (PHM) technique will be used for the estimation of the life cycle cost in relation to the maintenance activities of the machine. Many works have indicated that PHM techniques are used to reduce losses due to reliability issues (Pecht (2012)). Moreover, He et al. (2011) have successfully implemented the PHM techniques in the case of electromechanical migration on circuit boards, and more specifically, they proved that PHM techniques could be used in cases where the available physics-of-failure (PoF) models are unable to deliver satisfactory results, such as in Li-ion batteries and LEDs. Sun et al. (2012) have presented the challenges and benefits of the PHM techniques implementation and, more specifically, demonstrated that PHM techniques are based on the PoF approach, the data-driven approach, and the fusion approach. Toward this direction, Pecht and Jaai (2010) have reported that the physical understanding of the system failure mechanism, modeled mathematically, can determine the RUL. The main output of relevant research works has proved that machine learning techniques relying on the use of historical data can be classified as supervised and unsupervised learning techniques.

RECLAIM PHM will address the three main challenges concerning the refurbishment of a machine. First, is the machine worth refurbishing? Second, what is the best time to perform refurbishment at the least cost? Finally, how should the machine be refurbished? To determine if a machine is worth refurbishing, the RECLAIM implements well-established PHM techniques to estimate the life cycle cost associated with the refurbishment of the machine. If the cost of refurbishment is lower than that of a new machine, then it is a cost-effective option.

In RECLAIM, the optimal time to perform refurbishment is determined by extracting a machine health indicator through PHM and estimating the RUL of the machine based on the indicator. Ideally, the machine should be refurbished close to the end of its life. To determine how the machine should be refurbished, *in situ* edge-monitoring of PHM provides diagnosis information so that the degradation levels of critical components are estimated in near real-time. The outcome of edge analysis is combined with machine specifications and historical trends to create a trustworthy RECLAIM DSF that can bring benefits in terms of reduced costs and the environmental footprint of manufacturing activities.

Therefore, detailed knowledge of the health status of the machinery and its components and the whole production line based on data of the RECLAIM PHM toolkit will offer peer-to-peer health evaluation of the machine and component predictive methods to increase asset uptime; RECLAIM will be able to identify the optimal time and the appropriate components for refurbishment or/and remanufacturing.

2.1.5. Cost Analysis and Cost Modeling

Cost analysis and modeling cover the cost throughout the product life cycle, including design, manufacturing, service, and disposal. Manufacturing cost analysis and modeling have been well researched. Manufacturing costs normally include material costs, machining costs, and assembly costs, and the machining and assembly costs are normally calculated on the basis of the process design conducted by the production engineers (Xu et al. (2012)). The processing time can be determined based on the work rate of the resource used to perform each operation defined in the process. When the processing time is known, the machining or assembly costs can be estimated with the use of the cost rate of the resource utilization. For example, Xu et al. (2008a) adopted this approach to calculate the manufacturing costs in different applications, e.g., for aircraft life cycle cost modeling and automotive product manufacturing and remanufacturing cost modeling (Xu et al. (2014)). Maintenance costs are normally researched in the life cycle cost modeling covering different aspects. Xu et al. (2008b) developed aircraft life cycle cost modeling by using the Systems Engineering approach. In Xu and Feng (2014), a framework and cost model for the quantitative evaluation of the benefits of remanufacturing techniques to assist the decision-making on EoL strategies have been developed. Firstly, the additive manufacturing-based remanufacturing process has been modeled first; then, the cost breakdown structure for the process has been created; finally, the cost model has been developed.

Within RECLAIM, a real-time (or nearly real-time) accurate cost estimation model will be developed. This cost model will allow responsive and optimized decision-making regarding production and maintenance. This is achieved by conducting dynamic data collection based on the project's backbone big data infrastructure so that a real-time (or nearly real-time) accurate cost estimation can be carried out, which reflects the cost implications of real-time maintenance and projected disruptions to the production.

2.1.6. Optimization Planning for Refurbishment and Remanufacturing

Refurbishment and remanufacturing activities in a finite planning horizon are discussed thoroughly in the scientific literature. In the work of Ferrer (1997), the complexity of personal computing remanufacturing and the difficulties in developing adequate recovery processes are addressed. Guide and Wassenhove (2001) have devised a framework for profitability estimation of reuse activities demonstrating the way the product returns management influences operational requirements and stating that the acquisition of used products may be used as a leverage mechanism for the management and productivity of reuse activities. An overview of quantitative models taking into account logistic environment issues arising in the context of reverse logistics was given in Fleischmann et al. (1997). In Fleischmann et al. (1997), the authors have proposed a heuristic solution methodology based on a mathematical programming model for the reverse distribution problem. This approach is attributed to the complexity of the proposed model. Pirkul and Jayaraman (1996) followed a similar approach, as they developed a mixed integer programming model to address the location issue with the objective to minimize the total transportation and distribution costs and the fixed costs for opening and operating plants and warehouses by employing Lagrangian relaxation to the model.

Although several researchers have suggested various remanufacturing models, fewer researchers have been able to address the problems in both business and strategic decision-making at the same time. In this project, some mathematical models for decision-making at both business and strategic levels in a remanufacturing environment are proposed. The functionality and the performance of the suggested models are thoroughly examined through computational experiments, simulation schemas of model parameters, and extensive data analysis from multimodal data sources.

2.1.7. Digital Twin Simulation for Machinery Fault Diagnosis

The first attempt to analyze the concept of "twin" was made by NASA's Apollo Project in which researchers simulated the aircraft's twin body in a real physical system (Rosen et al. (2015)). In this way, astronauts could remotely observe the ship's condition and make decisions for emergency situations (Wang et al. (2015); Negri et al. (2017)). Beyond this application of the solution, Tao et al. (2018) have demonstrated the usefulness of the digital twin model throughout a machine's life cycle enabling the prediction of potential machine failures and closed-loop optimization of the machine design, production, operation, and maintenance. Moreover, in the same work, the authors have proved that the digital twin model allows technicians to verify the appropriate machine operation and monitor machine productivity. More specifically, they have demonstrated that the digital twin model can be used to control possible changes in the production phase before they are implemented since it could also be used to monitor the machine in all its operating phases and to reprogram it when

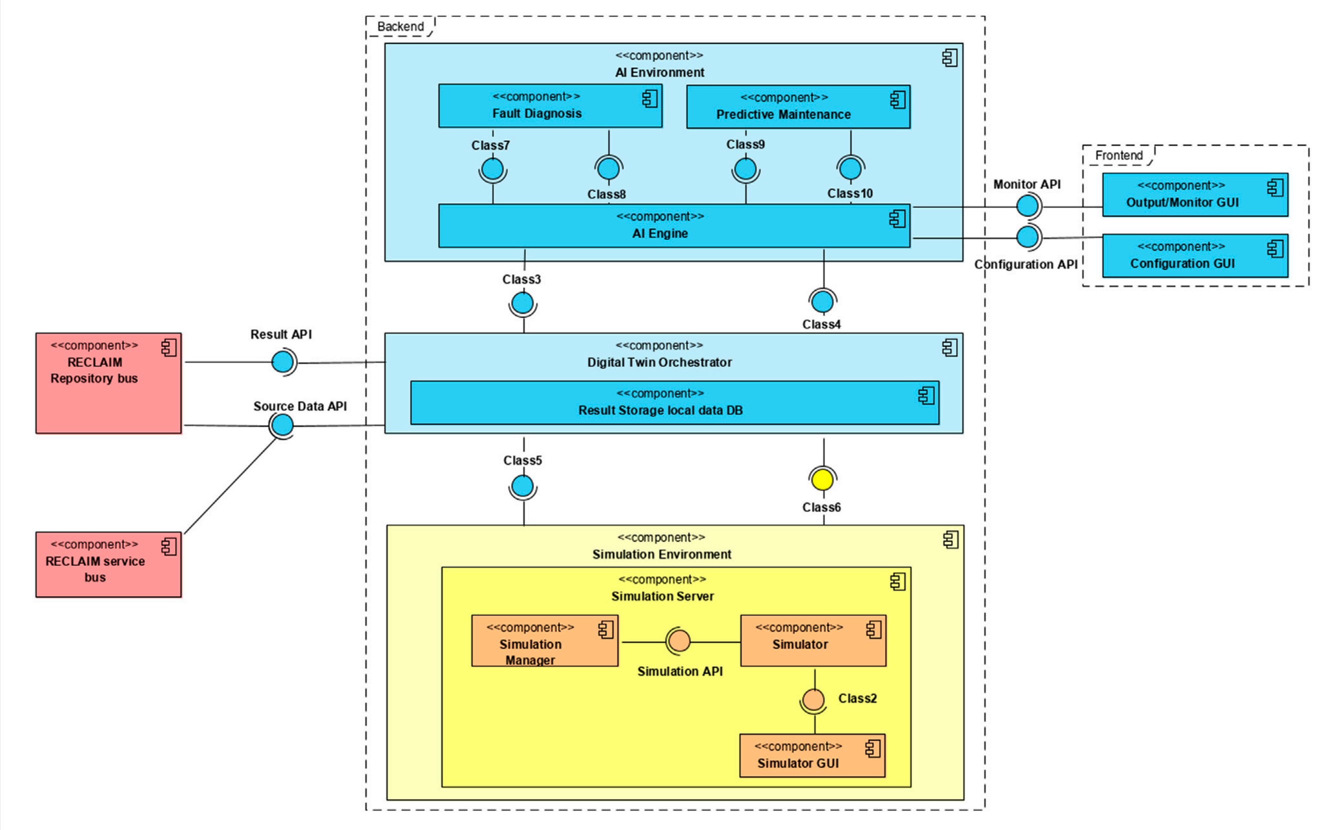


FIGURE 1 | Fault diagnosis and predictive maintenance simulation engine using digital twin.

necessary (e.g., from mass production to customized production). Finally, during the operation phase, the authors have affirmed that the digital twin model could be used as a verification tool for data collected by the factory on the performance of a product. Therefore, the digital twin model can offer value-added services with the support of physical simulation and data-driven intelligence. In more recent work, Magargle et al. (2017) have introduced a multiphysical twin model for monitoring the brake system status through multiple angles. To this end, the digital twin model can be implemented as a fault diagnostic service and as a RUL prediction and maintenance simulation tool, which can be utilized for prompt decisions and accident risk reduction.

In order to optimize maintenance activities (refurbishment, remanufacturing, etc.), RECLAIM will deploy a simulation engine that will create a virtual environment similar to the actual machine using the digital twin model. This model will analyze patterns in real-time and compare them with historical data about the machinery life cycle. The aim is to monitor and predict the performance and status of factory assets. In this manner, all the necessary information for the preparation and implementation of proper maintenance activities on the machines will be provided so that failures and production line stoppages are avoided. For this purpose, RECLAIM's digital twin model will be divided into three parts, i.e., the physical system (real industrial infrastructure), the digital system (simulated industrial infrastructure), and the data and information connection system that links the digital system with the physical system. RECLAIM's digital twin model saves time and money and helps reduce costly production downtimes as it predicts promptly possible failures in infrastructure.

Digital twin component, given in **Figure 1** includes the following subcomponents: a) the artificial intelligence (AI) environment, an engine leveraged to host and run the Fault Diagnosis and Predictive Maintenance algorithms; b) the AI engine, that is hosted in the AI environment and used to abstract the heterogeneous algorithms of Fault Diagnosis and Predictive Maintenance and to control their interactions; c) an orchestrator, that is used to orchestrate all tasks of the component, coordinating the interactions among AI engine and the distributed simulation environment, and receives the historical and real-time data, stores them, and processes them with data quality mechanisms; d) the simulation environment that is capable of running on different machines, each one wrapped by a simulation manager.

2.1.8. Cybersecurity for IoT Devices for Connected Smart Environments

Poor usability of cybersecurity solutions tends to be the effect of security constraints. Finding the right trade-off between usability and security or the preferable integration of usability and security requirements is part of a major research challenge, which recently has been addressed by scholars (Realpe et al. (2015)). For instance, user-centered approaches are recommended as means to accomplish usable security, while the definition of objectives for both security and usability is suggested as a way to decide on the right balance between the two (Dhillon et al. (2016)). Understanding the security and usability collectively is recognized as a critical factor for the successful development, implementation, and usage of information systems (Andriotis et al. (2016)). As far as the IoT is concerned, usability and security are identified as two of the four major research challenges (the other two being performance and reliability); privacy concerns are growing, as IoT device manufacturers for smart homes are acquired by large corporations, such as Google (Alur et al. (2016)). The most recent research suggests new usable security frameworks, particularly for modeling security and privacy risks in smart homes at the consumer level. For example, the framework presented in Nurse et al. (2016) aims to support home users with a highly usable security decision support tool. However, it still needs to address improvements on usability and scalability and validate the real utility offered to the user.

Within this project, the edge-computing capabilities of the proposed RECLAIM Solution will be enhanced with lightweight security methods in order to empower resilience to cyberattacks and intrusion detection and prevention. This will allow shared access and flexibility in data governance for edge-based applications and reconfiguration and actuation. Moreover, in order to ensure seamless and trusted service provision over different data, the RECLAIM Solution will be enhanced with capabilities related to the dynamic coupling of microservices offered and embedded devices involved.

2.1.9. Augmented Reality on the Plant Floor

AR is an emerging technology that can help manufacturers and maintainers, providing the necessary information which are needed regarding the maintenance/refurbishment/remanufacturing procedure by displaying virtual information on top of it. The main challenges faced by manufacturers or maintainers are as follows: (a) a large variety of tasks from diagnosis to repair; (b) varying complexity of maintenance requirements; (c) long life of equipment causing varying levels of quality, standards, and depth in documentation; (d) a large number of equipment types to maintain. It is noteworthy that AR offers opportunities for industrial maintenance applications by displaying contextualized information and accessing end-user data.

AR has received increasing attention from researchers in the manufacturing technology community as it is an interactive experience of a real-world environment and a technology that expands the physical world, adding on top of it layers of digital (virtual) information (Ong and Zhu (2013)). AR makes it possible for the user to gain information about a real-life process or procedure directly related to the work environment. This is the main coefficient factor for considering AR as an effective tool to be also used in through-life engineering services (Dini and Dalle (2015)). Several applications of AR in the industrial domain (maintenance, fault diagnosis, etc.) have been considered, but their research still remains at an exploratory level (Wang et al. (2016)). In order to study the effectiveness and the usability of AR integration in industrial fields (Oliveira et al. (2013)), new topics, such as authoring and context awareness, have arisen in this area (Erkoyuncu et al. (2017)). Authoring is a system component enabling the maintenance experts to develop, modify, and update applications' AR contents (Gimeno et al. (2013); Roy et al. (2016)), whereas context awareness is a system using context to provide to the user task-oriented information and/or services (Manuri (2016)). These main properties of AR focus on how information regarding maintenance is acquired, transformed, and presented to the process experts and maintainers so that they decide instantly about further maintenance steps and assure that the process will not be fatally disturbed in any way.

The RECLAIM project aims at using AR techniques to support maintenance operations in the industrial domain by creating augmented features in real-time. More specifically, the RECLAIM AR-enabled multimodal interaction system will address the above challenges as it will provide a novel way to visualize and localize information on equipment refurbishment and remanufacturing operations directly situated on top of the physical equipment. During the refurbishment and remanufacturing operations, the system will also provide animated 3D stepwise instructions on disassembly and reassembly required, as well as support in the form of on-the-job remote assistance with real-time audio-visual communication and 3D annotation to technicians during the procedure. In this way, the variety of tasks from diagnosis to repair is addressed by showing the steps that the technician must follow to repair or maintain the machine. Also, information about the useful lifetime of the machine will be displayed as pop-up messages. At the end of the repair, a function registration message will appear that will take into account the different quality levels, standards maintained, etc. One more functionality of this solution is that the system itself performs the authoring steps that require AR expertise and maintainers have to indicate the key information for display, defining its format and sequence. The Authoring Platform is a human-machine interaction interface that allows maintainers to interact with the "Information Frameworks" used for creating AR features. The Application Platform Modules automatically generate AR features according to maintainers' feedback. Apart from contextualizing and rendering maintenance information, they ensure that the information is displayed in the right sequence. The developed system is versatile and effective regarding the support of the maintenance work of both novice and more experienced technicians.

2.2. RECLAIM Platform Limitations

Despite the numerous advances of the RECLAIM Solution, some possible limitations have been considered. First of all, some pilots had not stored any related data before the beginning of the project, so training machine learning models or statistical analyses cannot be done based on many data from the past. This will raise the need for estimations based on information found in the bibliography about similar equipment. Also, probably some data (e.g., end-user actions) cannot be provided automatically by some hardware, so the end-users will have to interact with the RECLAIM platform to insert them manually. Furthermore, due to the complexity of interdependencies among software components and the algorithm underlying each of these components, it is possible for computational delays to occur due to the communication latency and/or time complexity of

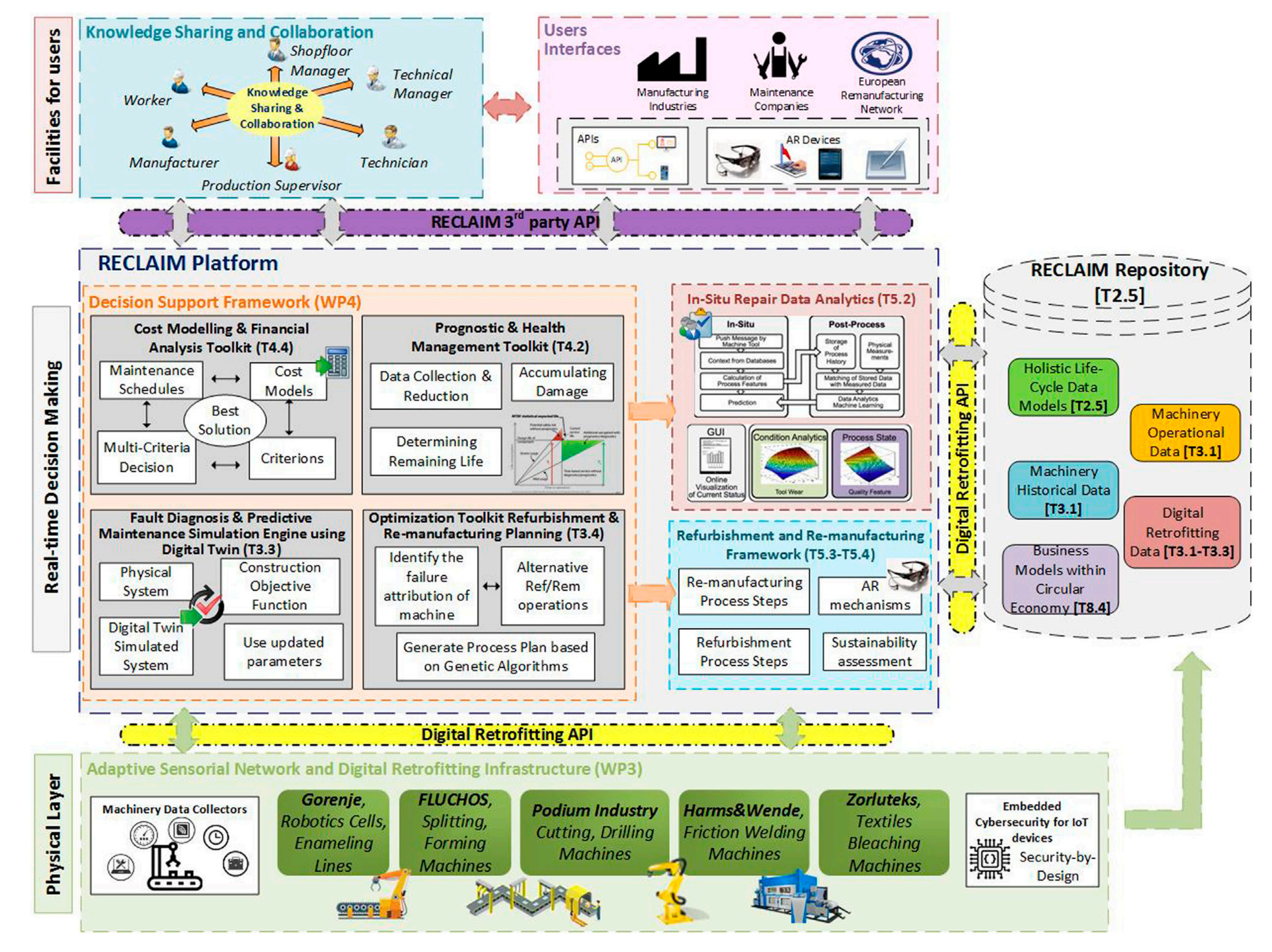


FIGURE 2 | RECLAIM conceptual architecture.

algorithms. In addition, the memory aspect should be taken into account. Given the plenty of pilot raw data, corresponding to several machines and their physical components, and output data from numerous software components, it may be challenging for a single database to store all of them.

2.3. RECLAIM Framework Validation

Several pilot sites across Europe will be used for demonstration, evaluation, and assessment activities of the RECLAIM project. Due to confidentiality issues, the pilot sites will not be named here. The RECLAIM framework will be tested on a white goods company for (a) refurbishment and renovation of robot cells in B-Cell for making tubs (they reach the end of their life and have a large number of failures in their operation, unexpected stoppages, and delays in the production line); (b) modernization and refurbishment of an automatic spraying cabin, using enamel powder applied with help of spraying guns, in a shoe-making factory for maintenance and upgrade of cutting machines (prognostics and health assessment and predictive maintenance or refurbishment or remanufacturing methodologies in terms of production system efficiency, production cost, and product quality), in a wood manufacturing plant for predictive maintenance and refurbishment of the woodworking large production line (refurbish the system with additional sensor capable of monitoring product quality and identify the cause of deviations and predict failures and breakdown(s)), in friction welding machinery for lifetime extension by remanufacturing and predictive maintenance (lifetime prediction, remanufacturing of the machine in order to meet the today's requirements, continuous online monitoring of the machine's state, and predictive maintenance features), and finally in a bleaching machine for maintenance, refurbishment, and upgrading (holistic core process parameters optimization, monitoring and process control tool, a safe and stable operation, and improving the resource efficiency).

3. CONCEPTUAL ARCHITECTURE

Figure 2 presents the envisioned architecture of RECLAIM. The integrated architecture encompasses several modules and components, which are described in the following sections. The physical layer of the conceptual architecture will be supported by several manufacturing pilots providing the necessary and on-point business scenarios on industrial machine remanufacturing, refurbishment, and maintenance. The IoT infrastructure will be based on a three-stage design, where stage 1 comprises the existing wireless sensors (monitoring, smart object, environmental, and legacy), stage 2 includes sensor data aggregation systems enabling analog-to-digital data conversion, and in stage 3, edge IT functions and modules perform preprocessing on multimodal data before moving on to the data center for the main processing. Furthermore, the machinery operational and historical data, along with life cycle and business models and digital retrofitting data, will be collected and stored at the data repository for use by the data analytics algorithms. The core component of the proposed architecture is the RECLAIM platform, a dedicated environment that contains three key components: the *DSF,* the *in situ repair data analytics*, and the *refurbishment and remanufacturing framework*. Finally, the topmost part of the proposed conceptual architecture is the *User(s) Interfaces* (UI), along with the *manufacturing industries*, the *maintenance supportive companies,* and the *European Remanufacturing Network* (ERN).

3.1. Architecture's Main Components Description

As mentioned above, the key components of the RECLAIM platform are the *DSF*, the *in situ repair data analytics*, and the *refurbishment and remanufacturing framework*, which are described in detail as follows.

3.1.1. Decision Support Framework

The *DSF* component is designed to support and improve the effectiveness of decisions concerning the refurbishment and remanufacturing of production infrastructure. The DSF will identify the most suitable actions/strategies based on different criteria such as the impact and the value of refurbishment or enhanced maintenance to extend asset life, the optimal time for replacing an asset, the machines' condition and possible upcoming failures, production planning, and resources allocation. The DSF will include novel tools as follows: the *Cost Modeling and Financial Analysis toolkit* (providing an effective cost estimation and financial impact analysis by using a combination of parametric and activity-based costing methods, while having the ability to take into account considerations of generality and reusability for the adaptation and uptake in wider industrial environments); the *Adaptive Sensorial Network and Fog Computing Framework* (providing information on the state of the machinery, such as temperature, operating speed, rotating speed, power consumption, torque, and vibration so as to minimize human interaction, increase mechanical automation, and identify pain points of machinery); the *Prognostic and Health Management toolkit* (providing a peer-to-peer health evaluation and component prediction methods to increase machine lifetime, productivity, and service quality); the *Fault Diagnosis and Predictive Maintenance Simulation Engine using Digital Twin* (creating a digital twin of the factory environment so as to use it to monitor and predict the performance and status of factory assets, while providing the user with all the features needed to schedule the maintenance works on the machines in order to avoid failures and to perform proper maintenance planning); the *Optimization Toolkit for Refurbishment and Remanufacturing Planning* (supporting the planning optimization through multivariable monitoring of the machine's operational parameters where the effects of variable changes will be possible to determine and combine best-known practices methodologies for model-based plant-site/shop-floor control). The proposed DSF will have attributes from both knowledge-driven and model-driven types of DSF based on the implementation of nondeterministic finite-state automata (finite number of states for the specified machine), simple scoring mechanisms, rule-based decision-making, and AI algorithms. Moreover, the implementation of data mining algorithms such as decision trees, genetic algorithms, and support vector machines will ensure the extraction of valuable information from IoT data. A *Visual Analytics Suite* will be built on the top of DSF in order to provide users with the most effective presentation of DSF output in the form of strategies, alternatives process models, KPIs visualization, and real-time health assessment of different production aspects.

3.1.2. *In Situ* Repair Data Analytics

Industrial analytics are used to identify and recognize machine operational and behavioral patterns, make fast and accurate predictions, and act with confidence at the points of decision. Situational awareness as a mental state can be considered as a state of knowledge, which can be achieved using various techniques. In particular, in order to raise awareness of the health status of the machine and the situation of the shop floor during maintenance activities, data analytics techniques can be deployed. So, the knowledge about the situation of machines or/and shop floor can be gained from descriptive analytics (gain insight from historical or current data streams), predictive analytics (creation of predictive models utilizing statistical and machine learning techniques for the identification of machine and processes behaviors), and prescriptive analytics (finding optimal solutions based on descriptive and predictive analytics aspects). In addition, big data methodologies (e.g., visual analytics and visualization techniques) for the analytics of the big and diverse volume of available data gathered by the industrial partners also can give some valuable information about situational awareness. Visual analytics and advanced information visualization technologies can be exploited to present relevant information to different users (shop-floor manager(s), technician(s), manufacturer(s), etc.) in a user-friendly and effective manner. In particular, visual analytics has historically played a key role in business processes optimization. Existing tools can be of great assistance for the visualization of spatiotemporal data in the plant-site/shop-floor, providing temporal plots and heat maps indicating specific types of activities and representation of process time series data along with statistical analysis suitable to assist in discovering variable correlation. The proposed visual analytics suite will be developed based on two major stages so as to support situational awareness: perception (monitoring), which refers to the knowledge of the elements in the environment of plant-site/shop-floor, and comprehension (inspection and exploring), which refers to the combination and the integration of elements received by the sensors network. In terms of on-site repair analytics, both streaming analytics from the field of the repair (e.g., work process of repair) and batch analytics results based on on-demand queries will be applied in a planning time horizon so as to ensure enhancing human decisions and understanding and generating significant confidence in the final decision.

3.1.3. Refurbishment and Remanufacturing Framework

Production planning, scheduling, and control of the fleet of an industrial ecosystem are major managerial challenges in the field of management operations in a manufacturing environment. A complete system approach is important to address all aspects of the production planning optimization, taking into account refurbishment and remanufacturing activities. This component aims to support the planning optimization through multivariable monitoring of the machine's operational parameters where the effects of variable changes will be possible to determine and combine best-known practices methodologies for model-based plant-site/shop-floor control. Based on the multimodal data provided by the IoT infrastructure, new approaches of real-time production planning optimization algorithms, from the perspective of machine learning techniques, will be researched and developed to apply proven optimization methodologies, provide the answers an end-user needs for effective decision-making, and consequently delivers measurable performance improvements. The data and information requirements are integral parts of the optimization phase. To create clear value from this information, production monitoring and surveillance is the first step in the measurement phase and is a prerequisite to analysis, improvement, and control. This monitoring might take into consideration the data collected from the *Adaptive Sensorial Network* together with recognition of any system constraints and behaviors.

3.2. Architecture's Core Innovations

RECLAIM is an ambitious project that will create and deploy an integrated DSF for machinery lifetime extension. The DSF for the optimization of refurbishment and remanufacturing process in itself is a significant step beyond the state of the art in the provisioning of infrastructure, tools, and services for experimentation in the digital manufacturing domain. The RECLAIM project will offer the following core innovations:

- *DSF* for refurbishment and remanufacturing optimization goes beyond the state of the art by allowing automatic and concurrent multiobjective (particularly for three or more objectives) decision-making and assessing multiobjectives in the same turn, so it provides efficient and optimized decision support.
- *In Situ Repair Data Analytics* for situational awareness will be a flexible tool, allowing the end-users to connect and integrate with any data source in real-time so as to have the ability for online visualization of significant KPIs for the current status of the machine (inspection) and the repair process. Condition analytics will be a key component that will combine state-of-the-art condition monitoring in order to go beyond machine health management.
- *Refurbishment and Remanufacturing Framework* will be enhanced with RECLAIM Solution having a sole objective; that is, the quality of remanufacturing/refurbishment process will have to follow strict reconditioning operations (steps). This effort will be supported by AR tools that utilize the sense of the worker, the ambient environment, and the context of work in the plant-site/shop floor.

4. CONCLUSION

The RECLAIM framework ensures that the remanufacturing and refurbishing interventions make a positive contribution not only toward business (i.e., increased return on investment, lifetime extension of the machinery, and alignment of its capabilities with the actual and future needs of the industry) but also toward the environment (i.e., improved material and resource efficiency and lower environmental impact). In particular, the proposed framework answers the following questions: *when* is the right time to refurbish or remanufacture industrial machinery, *what* is the appropriate strategy to follow, *which* benefits should the manufacturing company expect, and *how* this strategy will be implemented to deliver those benefits while providing enhanced reliability and safety of the refurbished or remanufactured equipment. The advanced RECLAIM framework aims to assist the machinery operators and machinery manufacturers in making efficient EoL decisions at different service life periods. This framework will consist of the following: a) the core RECLAIM toolkit (e.g., a Cost Modeling and Financial Analysis toolkit, a Prognostic and Health Management toolkit, a Fault Diagnosis and Predictive Maintenance Simulation Engine using Digital Twin, an Optimization Toolkit for Refurbishment and Remanufacturing Planning, and an *In Situ* Repair Data Analytics for Situational Awareness); b) a reference architecture along with a set of circular economy strategies and methodologies for manufacturing companies and OEMs.

RECLAIM is an ambitious project that will create and deploy an integrated DSF for machinery lifetime extension. The DSF for optimization refurbishment and remanufacturing process in itself is a significant step beyond the state of the art in the provisioning of infrastructure, tools, and services for experimentation in the digital manufacturing domain. The RECLAIM project will offer the following key novelty aspects: (a) a DSF for refurbishment and remanufacturing optimization by allowing automatic and concurrent multiobjective (particularly for three or more objectives) decision-making and assessing multiobjectives in the same turn, so it provides efficient and optimized decision support; (b) a Fault Diagnosis and Predictive Maintenance Simulation Engine using Digital Twin to keep the virtual twins that could store all available information during the lifetime of a machine, such as maintenance operations, and this information can be used to perform an economic estimation of the machine's refurbishment costs; (c) *In Situ Repair Data Analytics for Situational Awareness* that allows the end-users to connect and integrate with any data source in real-time so as to have the ability for online visualization of significant KPIs for the current status of the machine (inspection) and the repair process. Therefore, RECLAIM could answer *when* is the right time to refurbish/remanufacture industrial machinery, *what* is the appropriate strategy to follow, *which* benefits should the manufacturing company expect, and *how* this strategy will be implemented to deliver those benefits while providing for enhanced reliability and safety of the refurbished/remanufactured equipment. Through the proposed framework (DSF and associated tools, methodologies, and services), RECLAIM ensures that the remanufacturing and refurbishing interventions make a positive contribution not only businesswise (i.e., increased return on investment, lifetime extension of the machinery, and alignment of its capabilities with the actual and future needs of the industry) but also toward the environment (i.e., improved material and resource efficiency and lower environmental impact).

The next step after the implementation of RECLAIM is the validation of the proposed technology and the DSF through demonstrations. Those demonstrations will focus on large industrial equipment (e.g., industrial robotic systems, machines, AND production lines) from distinct industrial sectors: footwear manufacturers, white goods (cookers, dishwashers, etc.) manufacturers, wood manufacturing, friction welding machines, and textile manufacturers. Pilots will be supported by existing circular economy methods, which will be classified according to the preferable operation mode.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/Supplementary Material, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

AZ, conceptual architecture; TV, conceptual architecture and fault detection techniques; NK, optimization algorithms and fault detection techniques; YX, fault detection techniques; MP, conceptual architecture; DI, conceptual architecture; DT, conceptual architecture.

FUNDING

This work has been partially supported by the European Commission through the project HORIZON 2020-INNOVATION ACTIONS (IA)-869884-RECLAIM. The opinions expressed in this article are those of the authors and do not necessarily reflect the views of the European Commission.

REFERENCES

Alur, R., Berger, E., Drobnis, A. W., Fix, L., Fu, K., Hager, G. D., et al. (2016). Systems computing challenges in the internet of things. Available at: https://arxiv.org/pdf/1604.02980 (Accessed September 22 2015).

Andriotis, P., Oikonomou, G., Mylonas, A., and Tryfonas, T. (2016). A study on usability and security features of the android pattern lock screen. *Inf. Comput. Secur*. 24, 53–72. doi:10.1108/ICS-01-2015-0001.

Atasu, A., Sarvary, M., and Wassenhove, L. N. V. (2008). Remanufacturing as a marketing strategy. *Manag. Sci*. 54, 1731–1746. doi:10.1287/mnsc.1080.0893.

Chari, N., Diallo, C., and Venkatadri, U. (2014). State of the art on performability across the sustainable value chain. *Int. J. Perform. Eng*. 10, 543–556.

Cunha, V. P., Balkaya, I., Palacios, J., Rozenfeld, H., and Seliger, G. (2011). *Development of technology roadmap for remanufacturing oriented production equipment*. London: Springer.

Darghouth, M. N., Chelbi, A., and Ait-kadi, D. (2017). Investigating reliability improvement of second-hand production equipment considering warranty and preventive maintenance strategies. *Int. J. Prod. Res*. 55, 1–19. doi:10.1080/00207543.2016.1277277

Dehghanbaghi, M., Hosseininasab, H., and Sadeghieh, A. (2016). A hybrid approach to support recovery strategies (a case study). *J. Clean. Prod*. 113, 717–729. doi:10.1016/j.jclepro.2015.11.064

Dhillon, G., Oliveira, T., Susarapu, S., and Caldeira, M. (2016). Deciding between information security and usability: developing value based objectives. *Comput. Hum. Behav*. 61, 656–666. doi:10.1016/j.chb.2016.03.068.

Dhouib, D. (2014). An extension of macbeth method for a fuzzy environment to analyze alternatives in reverse logistics for automobile tire wastes. *Omega* 42, 25–32. doi:10.1016/j.omega.2013.02.003

Dini, G., and Dalle, M. M. (2015). Applications of augmented reality techniques in through-life engineering services. *Procedia CIRP* 38, 14–23. doi:10.1016/j.procir.2015.07.044

EP (2019). *General principles of eu industrial policy*.

Erkoyuncu, J. A., del Amo, I. F., Mura, M. D., Roy, R., and Dini, G. (2017). Improving efficiency of industrial maintenance with context aware adaptive authoring in augmented reality. *Ann. CIRP* 66, 465–468. doi:10.1016/j.cirp.2017.04.006

Ferrer, G. (1997). The economics of personal computer remanufacturing. *Resour. Conserv. Recycl*. 21, 79–108. doi:10.1016/S0921-3449(97)00030-X

Fleischmann, M., Bloemhof-Ruwarrd, J. M., Dekker, R., Lann, E. V. D., Nunen, J. A. E. E. V., and Wassenhove, L. N. V. (1997). Quantitative models for reverse logistics: a review. *Eur. J. Oper. Res*. 103, 1–17.

Freiberger, S., Albrecht, M., and Käufl, J. (2011). Reverse engineering technologies for remanufacturing of automotive systems communicating via can bus. *J. Remanuf*. 1, 1–14. doi:10.1186/2210-4690-1-6

Geng, X., Gong, X., and Chu, X. (2016). Component oriented remanufacturing decision-making for complex product using dea and interval 2-tuple linguistic topsis. *Int. J. Comput. Intell. Syst*. 9, 984–1000. doi:10.1080/18756891.2016.1237195.

Gimeno, J., Morillo, P., Orduña, J. M., and Fernández, M. (2013). A new ar authoring tool using depth maps for industrial procedures. *Comput. Ind*. 64, 1263–1271. doi:10.1016/j.compind.2013.06.012

Guide, V. D., and Wassenhove, L. N. V. (2001). Managing product returns for remanufacturing. *Prod. Oper. Manag*. 10, 142–155. doi:10.1111/j.1937-5956.2001.tb00075.x.

Hatcher, G. D., Ijomah, W. L., and Windmill, J. F. (2013). Design for remanufacturing in China: a case study of electrical and electronic equipment. *J. Remanuf*. 3, 1–11. doi:10.1186/2210-4690-3-3.

He, X., Azarian, M. H., and Pecht, M. G. (2011). Evaluation of electrochemical migration on printed circuit boards with lead-free and tin-lead solder. *J. Electron. Mater*. 40, 1921–1936. doi:10.1007/s11664-011-1672-3.

Kerr, W., and Ryan, C. (2001). Eco-efficiency gains from remanufacturing: a case study of photocopier remanufacturing at fuji xerox Australia. *J. Clean. Prod*. 9, 75–81.

Kremer, G. E. O. (2015). A fuzzy logic-based approach to determine product component end-of-life option from the views of sustainability and designer's perception. *J. Clean. Prod.* 108, 289–300. doi:10.1016/j.jclepro.2015.08.029.

Magargle, R., Johnson, L., Mandloi, P., Davoudabadi, P., Kesarkar, O., Krishnaswamy, S., et al. (2017). A simulation-based digital twin for model-driven health monitoring and predictive maintenance of an automotive braking system. Proceedings of the 12th international modelica conference, Prague, Czech Republic. May 15–17, 2017. Elsevier, 35–46. doi:10.3384/ECP1713235.

Manuri, F. (2016). A survey on applications of augmented reality. *Advances in CS* 5, 18–27.

M.NET (2016). *Predictive maintenance: marrying safety with productivity*.

Mori, M., and Fujishima, M. (2013). Remote monitoring and maintenance system for cnc machine tools. *Procedia CIRP* 12, 7–12. doi:10.1016/j.procir.2013.09.003

Mourtzis, D., Doukas, M., and Vandera, C. (2014b). Mobile applications for product customization and design of manufacturing networks. *Manuf. Lett.* 2, 30–34. doi:10.1016/j.mfglet.2014.01.002

Mourtzis, D., Doukas, M., Vlachou, A., and Xanthopoulos, N. (2014a). Machine availability monitoring for adaptive holistic scheduling: a conceptual framework for mass customization. *Procedia CIPR* 25, 406–413. doi:10.1016/j.procir.2014.10.056

Negri, E., Fumagalli, L., and Macchi, M. (2017). A review of the roles of digital twin in cps-based production systems. *Procedia Manufacturing* 11, 939–948. doi:10.1016/j.promfg.2017.07.198

Ng, Y. T., and Song, B. (2015). *Product characteristic based method for end-of-life product recovery*. London: Springer.

Nurse, J. R. C., Reineh, A.-A., and Martin, A. (2016). "Toward a usable framework for modeling security and privacy risks in the smart home," in Conference: International Conference on Human Aspects of Information Security, Privacy, and Trust, Vancouver, BC, July 9–14, 2017 (Springer), 255–267.

Oliveira, A., de Araujo, R. B., and Jardine, A. K. (2013). A human centered view on e-maintenance. *Chem. Eng. Trans.* 33, 385–390. doi:10.3303/CET1333065.

Ondemir, O., and Gupta, S. M. (2014). Quality management in product recovery using the internet of things: an optimization approach. *Comput. Ind.* 65, 491–504. doi:10.1016/j.compind.2013.11.006.

Ong, S. K., and Zhu, J. (2013). A novel maintenance system for equipment serviceability improvement. *Ann. CIRP* 61, 39–42. doi:10.1016/j.cirp.2013.03.091

Ovchinnikov, A., Blass, V., and Raz, G. (2014). Economic and environmental assessment of remanufacturing strategies for product + service firms. *Prod. Oper. Manag.* 23, 744–761. doi:10.1111/poms.12070

Pecht, M. G., and Jaai, R. (2010). A prognostics and health management roadmap for information and electronics-rich systems. *Microelectron. Reliab.* 50, 317–323. doi:10.1016/j.microrel.2010.01.006

Pecht, M. G. (2012). Nvidia's gpg failures: a case for prognostics and health management. *Microelectron. Reliab.* 52, 953–957. doi:10.1016/j.microrel.2011.11.017

Pirkul, H., and Jayaraman, V. (1996). Production, transportation, and distribution planning in a multi-commodity tri-echelon system. *Transport. Sci.* 30, 291–303. doi:10.1287/trsc.30.4.291

Realpe, P. C., Collazos, C. A., Hurtado, J. A., and Granollers, T. (2015). Toward an integration of usability and security for user authentication. *Resour. Conserv. Recycl.* 1–6.

Rehorn, A. G., Jiang, J., and Orban, P. E. (2005). State-of-the-art methods and results in tool condition monitoring: a review. *Int. J. Adv. Manuf. Technol.* 26, 693–710. doi:10.1007/s00170-004-2038-2.

Remery, M., Mascle, C., and Agard, B. (2012). A new method for evaluating the best product end-of-life strategy during the early design phase. *J. Eng. Des.* 23, 419–441. doi:10.1080/09544828.2011.605061.

Rosen, R. G., von Wichert, G. L., and Bettenhausen, K. D. (2015). About the importance of autonomy and digital twins for the future of manufacturing. *IFAC-PapersOnLine* 48, 567–572. doi:10.1016/j.ifacol.2015.06.141

Roy, R., Stark, R., Tracht, K., Takata, S., and Mori, M. (2016). Continuous maintenance and the future – foundations and technological challenges. *Annals of the CIRP* 65, 667–688. doi:10.1016/j.cirp.2016.06.006

Schraven, M., Heyer, S., and Rütthard, N. (2012). *Remanufacturing and reuse of production equipment at an automotive oem*. London: Springer.

Sharma, V., Garg, S., and Sharma, P. (2015). Remanufacturing process: the case of heavy equipment support services. *Int. J. Serv. Oper. Manag.* 22, 40–59. doi:10.1504/IJSOM.2015.070882.

Si, X.-S., Wang, W., Hu, C.-H., and Zhou, D.-H. (2011). Remaining useful life estimation – a review on the statistical data driven approaches. *Eur. J. Oper. Res.* 213, 1–14. doi:10.1016/j.ejor.2010.11.018.

SteingrímssonPinar, J. G., Bilge, P., Heyer, S., and Seliger, G. (2011). *Business strategies for competition and collaboration for remanufacturing of production equipment*. London: Springer.

Studios, W. C. (2018). *How manufacturers achieve top quartile performance*.

Sun, B., Zeng, S., Kang, R., and Pecht, M. G. (2012). Benefits and challenges of system prognostics. *IEEE Trans. Reliab.* 61, 323–335. doi:10.1109/TR.2012.2194173.

Tao, F., Cheng, J., Qi, Q., Zhang, M., Zhang, H., and Sui, F. (2018). Digital twin-driven product design, manufacturing and service with big data. *Int. J. Adv. Manuf. Technol.* 94, 3563–3576. doi:10.1007/s00170-017-0233-1

Tapoglou, N., Mehnen, J., Vlachou, A., Doukas, M., Milas, N., and Mourtzis, D. (2015). Cloud based platform for optimal machining parameter selection based on function blocks and real time monitoring. *J. Manuf. Sci. Eng.* 137, 040909. doi:10.1115/1.4029806.

Teti, R., Jemielniak, K., O'Donnell, G., and Dornfeld, D. (2010). Advanced monitoring of machining operations. *CIPR Ann.* 59, 717–739. doi:10.1016/j.cirp.2010.05.010

Ullah, S. M. S., Muhammad, I., and Ko, T. J. (2016). Optimal strategy to deal with decision making problems in machine tools remanufacturing. *Int. J. Precision Eng. Manuf. Green Technol.* 3, 19–26. doi:10.1007/s40684-016-0003-9.

Varde, P. V., Tian, J., and Pecht, M. G. (2014). "Prognostics and health management based refurbishment for life extension of electronic systems," in International conference on information and automation (ICIA), Lijiang, China, July 28–30, 2014 (IEEE), 1260–1267.

Wang, P., Gao, R. X., and Fan, Z. (2015). Cloud computing for cloud manufacturing: benefits and limitations. *J. Manuf. Sci. Eng.* 137, 040901. doi:10.1115/1.4030209.

Wang, X., Ong, S. K., and Nee, A. Y. C. (2016). A comprehensive survey of augmented reality assembly research. *Adv. Manuf.* 4, 1–22. doi:10.1007/s40436-015-0131-4.

Wollenhaupt, G. (2017). *Iot slashed downtime with predictive maintenance*.

Xu, Y., Elgh, F., Erkoyuncu, J. A., Bankole, O., Goh, Y. M., Cheung, W. M., et al. (2012). Cost engineering for manufacturing: current and future research. *Int. J. Comput. Integrated Manuf.* 25, 300–314. 10.1080/0951192X.2010.542183.

Xu, Y., and Feng, W. (2014). Develop a cost model to assess the economic benefit of remanufacturing. *J. Remanuf* 4, 4. doi:10.1186/2210-4690-4-4.

Xu, Y., Sanchez, J. F., and Njuguna, J. (2014). Cost modeling to support optimized selection of end-of-life options for automotive components. *Int. J. Adv. Manuf. Technol.* 73, 399–407. doi:10.1007/s00170-014-5804-9.

Xu, Y., Wang, J., Tan, X., Curran, R., Raghunathan, S., Doherty, J., et al. (2008a). "Manufacturing cost modeling for aircraft wing," in The 6th International conference on manufacturing research (ICMR08), London, United Kingdom, September 9–11, 2008.

Xu, Y., Wang, J., Tan, X., Raghunathan, S., Doherty, J., and Gore, D. (2008b). "A generic life cycle cost modeling approach for aircraft system," in *Collaborative product and service life cycle management for a sustainable world*. Springer, 251–258.

Ziout, A., Azab, A., and Atwan, M. (2014). A holistic approach for decision on selection of end-of-life products recovery options. *J. Clean. Prod.* 65, 497–516. doi:10.1016/j.jclepro.2013.10.001.

Conflict of Interest: Author MP is employed by the company Harms and Wende GmbH and Co.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Advantages of publishing in Frontiers

OPEN ACCESS
Articles are free to read for greatest visibility and readership

FAST PUBLICATION
Around 90 days from submission to decision

HIGH QUALITY PEER-REVIEW
Rigorous, collaborative, and constructive peer-review

TRANSPARENT PEER-REVIEW
Editors and reviewers acknowledged by name on published articles

Frontiers
Avenue du Tribunal-Fédéral 34
1005 Lausanne | Switzerland

Visit us: www.frontiersin.org
Contact us: frontiersin.org/about/contact

REPRODUCIBILITY OF RESEARCH
Support open data and methods to enhance research reproducibility

DIGITAL PUBLISHING
Articles designed for optimal readership across devices

FOLLOW US
@frontiersin

IMPACT METRICS
Advanced article metrics track visibility across digital media

EXTENSIVE PROMOTION
Marketing and promotion of impactful research

LOOP RESEARCH NETWORK
Our network increases your article's readership